University of Plymouth

Guidance and Control of an Autonomous Underwater Vehicle

Wasif Naeem

A thesis submitted for the degree of

Doctor of Philosophy

in the

Faculty of Technology
School of Engineering

In Collaboration with the J&S Marine Ltd., QinetiQ,
Subsea 7 and South West Water PLC

September 2004

To my parents ....

University of Plymouth

Guidance and Control of an Autonomous Underwater Vehicle

Wasif Naeem

Abstract

A cooperative project between the Universities of Plymouth and Cranfield was aimed at designing and developing an autonomous underwater vehicle named *Hammerhead*. The work presented herein is to formulate an advance guidance and control system and to implement it in the *Hammerhead*. This involves the description of *Hammerhead* hardware from a control system perspective. In addition to the control system, an intelligent navigation scheme and a state of the art vision system is also developed. However, the development of these submodules is out of the scope of this thesis.

To model an underwater vehicle, the traditional way is to acquire painstaking mathematical models based on laws of physics and then simplify and linearise the models to some operating point. One of the principal novelties of this research is the use of system identification techniques on actual vehicle data obtained from full scale in water experiments. Two new guidance mechanisms have also been formulated for cruising type vehicles. The first is a modification of the proportional navigation guidance for missiles whilst the other is a hybrid law which is a combination of several guidance strategies employed during different phases of the flight.

In addition to the modelling process and guidance systems, a number of robust control methodologies have been conceived for *Hammerhead*. A discrete time linear quadratic Gaussian with loop transfer recovery based autopilot is formulated and integrated with the conventional and more advance guidance laws proposed. A model predictive controller (MPC) has also been devised which is constructed using artificial intelligence techniques such as genetic algorithms (GA) and fuzzy logic. A GA is employed as an online optimization routine whilst fuzzy logic has been exploited as an objective function in an MPC framework. The GA-MPC autopilot has been implemented in *Hammerhead* in real time and results demonstrate excellent robustness despite the presence of disturbances and ever present modelling uncertainty. To the author's knowledge, this is the first successful application of a GA in real time optimization for controller tuning in the marine sector and thus the thesis makes an extremely novel and useful contribution to control system design in general. The controllers are also integrated with the proposed guidance laws and is also considered to be an invaluable contribution to knowledge. Moreover, the autopilots are used in conjunction with a vision based altitude information sensor and simulation results demonstrate the efficacy of the controllers to cope with uncertain altitude demands.

Acknowledgements

In the beginning, I must thank Allah, the Almighty and the Most Merciful for His blessings throughout my life in general and in the course of this thesis in particular.

I am deeply indebted to my Director of Studies, Prof. Robert Sutton, who gave me the opportunity to pursue my research work within his group. His help and encouragement throughout my course made me feel not far away from home. His constant support and guidance assisted me in improving many of my shortcomings and of course his red pen provided great help in ameliorating my technical report writing skills.

I would also like to express my gratitude to my co-supervisor Dr John Chudley whose friendly character always encouraged me to speak out. There are a great many people here in Plymouth which I would like to thank for all their support throughout my PhD and to make my stay here a memorable one. First of all, my colleague and project mate, Dedy Loebis, whose seriousness and devotion to work is undoubtful. A big thanks to Mike Sloman, the mechanical engineering department technician, for fabricating the test rig and providing a helping hand during the *Hammerhead* trials. I am also extremely thankful to Barbara Fuller who took the pain to help me sort out the departmental matters, and my adventuresome journeys to some local and overseas conferences. Thanks are extended to Dr Steve Tetlow and Fraser Dalgleish at the Offshore Technology Centre, Cranfield University, for their utmost cooperation in conducting trials with the *Hammerhead* AUV. Thanks are also due to EPSRC for the overall funding of this project.

I would like to pay a high tribute to my family, including my parents, brother, sister, brother-in-law and sister-in-law, for their continuing love, support and sacrifices throughout my career. Special thanks goes to my fiancee, Amara, for her love and patience; we are going to be together very soon. Finally thanks goes to the unwavering accompaniment of my uncle and aunt here in the UK.

Last but not the least, I am very thankful to all my friends I made here in Plymouth for their outstanding support throughout the duration of my research work. I am truly indebted to all of them.

Declaration

This is to certify that

- the thesis comprises only my original work towards the PhD,
- due acknowledgement has been made in the text to all other material used,
- during the candidature, I have not been registered for any other award at any other institution,
- this study was financed with the aid of a studentship from the Engineering and Physical Sciences Research Council, UK,
- relevant scientific seminars and conferences were regularly attended at which work was often presnted; external institutions were visited for consultation purposes and several papers prepared for publication,
- the thesis is 49,995 words in length, exclusive of tables, bibliographies and appendices.

Wasif Naeem
15^{th} September 2004

Table of Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

ACF	Autocorrelation function
AI	Artificial intelligence
ARMAX	Auto-regressive moving average with exogeneous input
ARX	Auto-regressive with exogeneous input
AUV	Autonomous underwater vehicle
CCF	Cross correlation function
COA	Circle of acceptance
DAQ	Data acquisition
DMT	Deep mobile target
DOF	Degrees of freedom
DVL	Doppler velocity log
FOV	Field of view
FPE	Final prediction error
GM	Gain margin
GPS	Global positioning system
IMU	Inertial measurement unit
KMN	Moments about the x, y and z directions respectively
LBL	Long base line
LOS	Line of sight
LQG	Linear quadratic Gaussian

LQR	Linear quadratic regulator
LTR	Loop transfer recovery
MPC	Model predictive control
MSDF	Multisensor data fusion
MSE	Mean square error
NGC	Navigation, guidance and control
PID	Proportional integral and derivative
PM	Phase margin
PNG	Proportional navigation guidance
PRBS	Pseudo random binary sequence
QP	Quadratic programming
ROV	Remotely operated vehicles
SBL	Short base line
SISO	Single input single output
STD	Standard deviation
UUV	Unmanned underwater vehicle
XYZ	Forces in the x, y and z directions respectively

Symbols

α	Course change of the vehicle in a given time (deg)
δ	Impulse function
$\dot{\lambda}$	Line of sight angle rate (deg/s)
η	Vector of position and Euler angles
η_c	Missile acceleration (m/s^2)
Γ	Model coefficients
$\hat{y}$	Model output
λ	Line of sight angle (deg)
T	Transformation matrix

τ	Forces and moments acting on the vehicle in the body-fixed coordinate system
τ_E	Environmental forces and moments acting on the vehicle
τ_H	Hydrodynamic forces and moments acting on the vehicle
τ_{RB}	Vector of external forces and moments about the origin acting as an input to the rigid body
$\mathbf{C}_A$	Coriolis and centripetal effects due to added mass
$\mathbf{C}_{RB}$	Coriolis and centripetal forces acting on the rigid body
$\mathbf{D}(\nu)$	Hydrodynamic damping matrix
$\mathbf{M}_A$	Added mass matrix
$\mathbf{M}_{RB}$	Rigid body inertia matrix
$\mathbf{Q}$	State weighting matrix
$\mathbf{R}_v$	Measurement noise covariance matrix
$\mathbf{R}_w$	Process noise covariance matrix
ν	Vector of linear and angular velocities
ϕ	Roll of the vehicle in earth fixed reference frame (deg)
ψ	Yaw or heading of the vehicle in earth fixed reference frame (deg)
ρ	Distance of vehicle from the waypoint (m)
ρ_o	Radius of the circle of acceptance (m)
θ	Pitch of the vehicle in earth-fixed reference frame (deg)
ϵ	Residuals
D^n	Number of data points in nth data set
g	Vector of gravitational and buoyant generalised forces
I_x	Moment of inertia about the x-axis
I_y	Moment of inertia about the y-axis
I_z	Moment of inertia about the z-axis
I_{xy}	Product of the inertia about x and y-axis
I_{xz}	Product of the inertia about x and z-axis

I_{yz}	Product of the inertia about y and z-axis
J	Performance index
K_c	LQR state feedback gain matrix
K_f	Kalman filter gain matrix
L	Straight line distance travelled by the vehicle (m)
m	Mass of the vehicle (kg)
N'	Navigation ratio
p	Angular velocity in the x-direction (deg/s)
q	Angular velocity in the y-direction (deg/s)
R	Input weighting matrix
r	Angular velocity in the z-direction (deg/s)
r_a	Turning radius of the vehicle (m)
S	Arc length or surface distance of a circle (m)
t_f	Interception time (s)
u	Linear velocity in the x-direction (m/s)
v	Linear velocity in the y-direction (m/s)
V_c	Closing velocity (m/s)
w	Linear velocity in the z-direction (m/s)
x_G	x-coordinate of the centre of gravity of the vehicle (m)
y_G	y-coordinate of the centre of gravity of the vehicle (m)
y_n	Measurement of the nth data set
z_G	z-coordinate of the centre of gravity of the vehicle (m)

Chapter 1

Introduction

1.1 Motivation

The advancement of technology, like population growth has been proceeding exponentially. A major source of incentive for technological development lies in the demands of time. Systems are getting much more complex and the applications are also diversified. This rapidly changing environment provides the technologist with a multifaceted task to excel and find feasible solutions to a problem at hand. Underwater vehicle technology is no exception.

Designing underwater vehicles present enormous challenges to engineers. This is mainly due to the hostile underwater environment and the degrees of freedom of the vehicle movement. The last decade has seen a tremendous boost in the underwater vehicle development for exploring the rich and untapped underwater world containing a huge number of natural resources. As the oceanographer, James Gardner says as quoted by Barber (2001)

> "*We know what the surface of the moon is better than we know what the surface of the sea floor is.*"

Clearly this gives a hint that there is still a considerable amount of research work to be done in order to explore deep oceans which cover more than 70% of the Earth. The main hurdle in deep sea exploration is the inability of human divers to reach those places. Underwater vehicles are thought to be a true replacement of deep sea divers for ocean surveying. In addition, they are being used in covert missions and for mine clearing operations as the world has recently witnessed by the use of

Hydroid's *REMUS* (Remote Environmental Monitoring Unit) underwater vehicle in the Iraqi conflict of 2003 (Jordan, 2003). Perhaps one of the main impetus for the design of state of the art underwater vehicles evolved in 1986 after the discovery of the *Titanic* wreckage when the Woods Hole Oceanographic Institution's (WHOI's) *Jason Jr.* underwater vehicle was seen at the wreck site shown in Figure 1.1 probing into and taking photographs from inside the ship (Carlowicz, 2003).

Figure 1.1: *Jason Jr.* underwater vehicle peers into a window on *Titanic* (courtesy of WHOI website `www.whoi.edu/home`) (Carlowicz, 2003)

To date, a significant amount of work has already been undertaken by different research groups around the world on underwater vehicle design and there exists a plethora of configurations and several navigation, guidance and control (NGC) laws governing their operation. Underwater vehicles are generally classified into two types, namely, remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs). Common to these two types of vehicle is the fact that both ROVs and AUVs are unmanned i.e. they do not take any human onboard but instead are a collection of sensors, actuators and thrusters. Therefore, they can operate in deep oceans and in hazardous areas such as near underwater volcanoes or in a mine hunting mission without jeopardising human life which is a primary source of inducement for their use. Thus relieving all those involved from the liability of exposing humans to undersea hazards.

An AUV has an onboard navigation, guidance and control system and it only requires time to time monitoring and supervision or no supervision at all. On the other hand, an ROV is attached through a data and power tether to a support platform on the surface where a human operator constantly monitors its state and manoeuvres it accordingly. ROVs are generally equipped with a control system such as the *Falcon* by Seaeye Marine Ltd. (Seaeye, 2004), however, they rely heavily on the human operator guidance. It is obvious that the ROV mission duration is limited by the length of the tethered cable and the cost of a mission is high due to the requirement of a trained ROV operator and a surface support platform. On the contrary, the AUV mission duration is only confined by the onboard power generating system.

Following a succinct overview of AUVs and ROVs, the aim of this thesis alone is to design an advance guidance and control system of an AUV named *Hammerhead* shown in Figure 1.2 developed as a joint effort by the Universities of Plymouth and Cranfield. The name *Hammerhead* was assumed from the fact that the nose of the vehicle resembles to that of a *Hammerhead* Shark because of the installation of front hydroplanes in addition to the camera (vision system) mounted near to them.

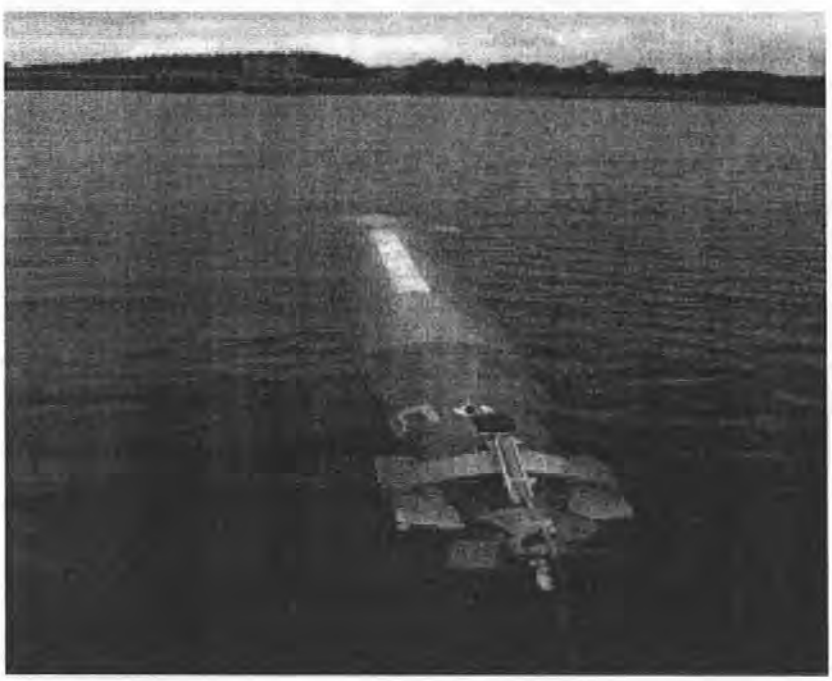


Figure 1.2: The torpedo shaped *Hammerhead* AUV during a test trial at Roadford Reservoir, Devon, UK

A multisensor data fusion (MSDF) based navigation module and a state of the art vision system is also conceived for the *Hammerhead*. However, the development of these submodules is out of the scope of this thesis. Some generic objectives of the *Hammerhead* project are provided below.

1.1.1 *Hammerhead* project objectives

Enlisting another AUV. An obvious outcome of the *Hammerhead* project is the enlisting of another AUV under UK's name. To date, the AUV market in the UK has grown significantly with several AUVs on the horizon such as the *AUDOS* AUV for tracking deep-sea scavenging fish by the University of Aberdeen (Jones *et al.*, 2004) and the multi-purpose *RAUVER* vehicle (Hamilton, 2004) and *AUTOTRACKER* (Ruiz, 2004) for cable/pipeline inspection by the Heriot Watt University. However, the two main noticeable AUV programmes in the UK are the Natural Environmental Research Council's (NERC) *AUTOSUB* project based at the Southampton's Oceanography Centre (Millard *et al.*, 1998) and the *Marlin* AUV (Tonge, 2000) developed by BAE systems for QinetiQ (formerly DERA). Developing *Hammerhead* provides Plymouth and Cranfield with a competitive edge over other underwater research groups within the UK and around the world.

Low cost factor. Both *AUTOSUB* and *Marlin* AUVs incurred very high cost with *Marlin* cost around £1.0 million (Wilks, 2000) and *AUTOSUB* stands at £5.30 million to bring it to its 1997 specification (Millard *et al.*, 1998). One of the aims of this project is to deliver a low cost AUV facility in the UK that is almost equally capable or even better than other vehicles of similar configuration.

Providing a test bed for other researchers. The fact that the *Marlin* AUV was developed for military purpose leaves *AUTOSUB* and others to serve the scientific underwater research community. Therefore, there is a strong urge to develop AUVs within the UK which are easily deployable and can accommodate various requirements without varying the existing payload on them. The *Hammerhead* vehicle fulfills this by having various commercial off the shelf navigational sensors onboard and a state of the art vision system, a forward looking sonar and control computers. Thus offering a practical solution in the form of a test bed for academic research.

1.1.2 Navigation, guidance and control

Guidance plays the key role in bringing autonomy to a system. The guidance system manipulates the sensor outputs and by utilising some guidance law generates suitable trajectories to be followed by the vehicle. This takes into account the target and any obstacles that may be encountered during the course of a flight. Once a trajectory has been decided, the control system takes charge and steers the vehicle in the appropriate direction by sending commands to the actuators. The control system must be robust enough to deal with unknown or unmeasured disturbances such as sea currents, sensors noise and model uncertainty.

In addition to the guidance and control system, a reliable navigation system is also imperative for true autonomous operation. A simple block diagram of an integrated NGC system is depicted in Figure 1.3.

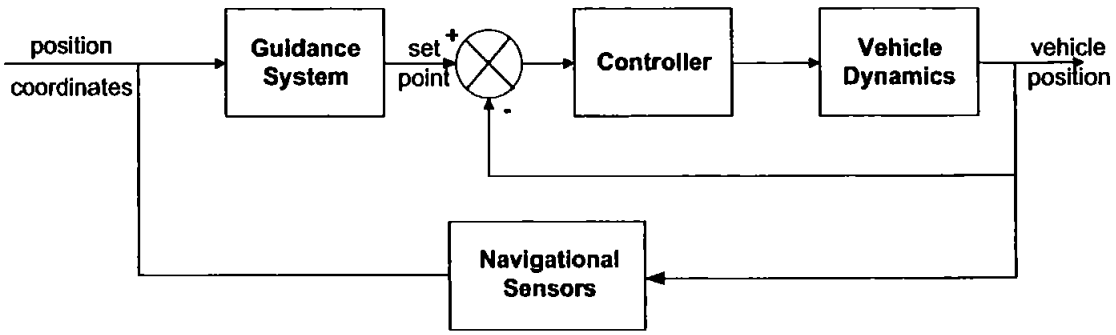


Figure 1.3: Navigation, guidance and control for a vehicle

The navigation system is responsible for accurate positioning of the AUV on the surface and in the water. By reliable, it means that it should be able to cope with any kind of sensor noise and also in cases where a fault occurs in one or more sensors. For this purpose, an MSDF algorithm is used which combines data coming from various onboard sensors and provides the best estimate of the actual state of the AUV. A good account of several data fusion techniques has been documented by Loebis *et al.* (2002). For the *Hammerhead*, an intelligent MSDF technique based on fuzzy logic and a multiobjective genetic algorithm has been proposed (Loebis *et al.*, 2004) and developed at the University of Plymouth. The MSDF based navigation system provides estimate of the vehicle position using a global positioning system (GPS) on the surface and dead reckoning while submerged. A laser stripe illumination based vision system is also developed by Dalgleish *et al.* (2002) for *Hammerhead* at Cranfield

University to further enhance the navigation system. The vision system is dominant and will become the primary guidance system when the object to be tracked is in the field of view (FOV) of the camera. However, if the vision system loses the subject or the water is too murky, then the guidance system relies heavily on the dead reckoning position estimates.

1.2 Aim and Objectives of the Research

The overall aim of this thesis is to develop an integrated guidance and control system for the *Hammerhead* vehicle. This would also entail the implementation of the controllers in real time. Broken down as submodules, the objectives of this research are provided as follows

a. To review various guidance laws for unmanned systems including missiles with an emphasis on underwater vehicles and to investigate the feasibility of guidance technology transfer from airborne vehicles to underwater robotic systems.

b. One of the mission objectives of the *Hammerhead* AUV is to follow a cable/pipeline for inspection purpose. Therefore development of several guidance algorithms to achieve this task will be undertaken.

c. To design and conduct experiments to produce data sets for yaw and depth channels suitable for *Hammerhead* model identification.

d. To develop linear transfer function and state space models for the yaw and depth channels using system identification techniques on the trials data for control systems design.

e. To investigate the cross coupling effects among different channels of *Hammerhead* such as yaw and roll.

f. To develop various optimal control laws and assess their performance on *Hammerhead* AUV in simulations for various mission scenarios.

g. To design and conduct the necessary experiments to evaluate the performance of controllers in full scale trials.

1.3 Thesis Overview

In recent years, control systems have assumed an increasingly important role in the development and advancement of modern civilisation and technology. In particular, the burgeoning in the field of NGC systems, spurred on mainly by the challenges of unsolved aerospace problems, contributed significantly to progress achieved in the development of modern systems and control theories. The success of the Soviet Union's satellite technology in the 1950s stimulated the United States to develop their own aerospace technology thus creating between the two of them new concepts in the field of control system design. The Apollo programme in the 1960s is a classical example of the translation of various NGC concepts into working modules. The early success of NGC systems soon led to advances in such diverse areas as industrial manufacturing, energy management (Lin, 1991) and underwater vehicles. Although applications of NGC in these areas have shown a profound impact in control theory in general, the majority of research and development in NGC continues to find its main application in the aerospace industry. Navigation, guidance and control of airborne systems have been reported extensively in the literature (Lin, 1991; Cloutier *et al.*, 1989; Lin and Su, 2000), however, little attention has been paid to the issue of guidance of unmanned underwater vehicles (UUVs).

A comprehensive review of the current scene on guidance of UUVs has thus been undertaken by Naeem *et al.* (2003*d*). Chapter 2 is the extract of this paper and presents some, if not all, of the guidance laws found in the literature. Missile system guidance is briefly reviewed followed by some well known guidance techniques used for UUVs such as waypoint following and vision based guidance for object tracking. The chapter also covers guidance technology transferred from airborne systems to underwater vehicles.

Since there is no previous hardware details of *Hammerhead* in the literature, it was deemed necessary to disseminate this information in Chapter 3. This includes the *Hammerhead* navigational suite, placement of sensors and the hardware/software that has been developed at Plymouth. The chapter also elaborates on how the *Hammerhead* hardware setup has evolved from an ROV type configuration (communicating through an umbilical) to an autonomous conformation.

Equipped with the knowledge of *Hammerhead* hardware, Chapter 4 describes the

modelling process. It is shown that system identification (SI) techniques gives reasonably accurate models in a short period of time in contrast to painstaking mathematical modelling involving differential equations and requiring tank tests to find hydrodynamic coefficients which are expensive and not available everywhere. Separate models for yaw and depth channels are obtained using SI on trials data and simulation results are presented for model validation.

Chapter 5 proposes two guidance laws for cable/pipeline tracking problem. The first one called the pure pursuit guidance has been derived from the well known *proportional navigation guidance* law for missile systems. The next is a hybrid law which utilises the vehicle's speed as a means to formulate the guidance strategy. It is shown that the line of sight (LOS) guidance is the key ingredient of all guidance laws. Simulation results are presented to demonstrate the trajectory planning capabilities of the proposed guidance strategies.

Having decided on the model of the *Hammerhead* and guidance system, attention is turned towards the design of a suitable controller. Chapter 6 explains the development of a linear quadratic Gaussian controller with loop transfer recovery (LQG/LTR) for the *Hammerhead* yaw and depth channels. The LQG/LTR controller is selected owing to its robustness and stability properties. Simulation results are presented for simple waypoint following including effects of sea currents, depth control and cable tracking mission by integrating it with the proposed guidance laws in Chapter 5. A vision based altitude information sensor is also integrated and results are shown for the case when the guidance commands are uncertain.

Extending the controller design of *Hammerhead*, Chapter 7 develops a model predictive controller (MPC) optimized using a genetic algorithm (GA). Two forms are proposed and developed for this class of controller. The first is using the conventional quadratic objective function whilst the other uses a fuzzy cost function. Simulation results are shown for both type of autopilots. By integrating the MPC with the pure pursuit guidance and hybrid guidance law from Chapter 5, cable tracking missions are performed in simulations and results are illustrated. The effect of sea currents on waypoint following is also investigated. Finally, simulation results are shown for an altitude control system when the sensor data is uncertain or noisy.

Chapter 8 covers experimental results from various *Hammerhead* trials during the

course of its development. In addition, the current hardware and software configuration of the vehicle is also elucidated. The results mainly consists of LOS tracking mission using the GA-MPC autopilot with and without the presence of the umbilical. The GA is shown to be running in real-time which to the author's knowledge has not been tested in the marine sector before. It is demonstrated that the controller performs remarkably well in real time environment despite the existence of disturbances and ever present modelling uncertainty.

Finally Chapter 9 contains the summary, conclusions of the work presented within this thesis and recommendations for any further work.

Chapter 2

A Review of Guidance Laws Applicable to Unmanned Underwater Vehicles

This chapter reviews a number of guidance laws, which have been adopted for the guidance of air and sea vehicles with an emphasis on UUVs. Guidance laws for the airborne missile systems are presented first followed by a detailed description of UUV guidance. In addition, the chapter explores ways and means of employing successful guidance strategies of air based systems to underwater vehicles. This would also entail certain modifications to suit the underwater mission requirement. It should be noted that this chapter is a modified version of the paper by Naeem *et al.* (2003*d*) which is provided in Appendix D.

2.1 Guidance

All autonomous vehicles must have onboard NGC systems, which should work in accord with each other for proper operation. Imperfections in one system degrade the efficiency of the other. The navigation system provides information related to the target, which is processed by the guidance system to generate reference headings. The control system is responsible for keeping the vehicle on course as specified by the guidance processor. In remotely operated systems, guidance commands are sent from a ground station by a trained human operator whilst autonomous vehicles have an onboard guidance processor. With regard to this, a guidance system plays the vital role in bringing autonomy to the system. Some definitions and a brief description of

the elements of a guidance system are now presented as follows

> "*guidance is the action of determining the course, attitude and speed of the vehicle, relative to some reference frame, to be followed by the vehicle*" (Fossen, 1994).

From the perspective of a control system,

> "*guidance is a matter of finding the appropriate compensation network to place in series with the plant in order to accomplish an intercept*" (Lin, 1991).

The guidance system decides the best trajectory (physical action) to be followed by a vehicle based on target location and vehicle capability. The primary function of the elements that constitute a guidance system are sensing, information processing and correction. Guidance issues are mainly determined by the nature and location of the target and the environmental conditions. The nature of the target corresponds to the condition as to whether or not the target is stationary, moving, or manoeuvring. The target location is also imperative as it determines the heading and depth to be followed by the vehicle, however, the accuracy of the system depends on the environmental conditions. The guidance problem is also related closely to the bandwidth of the system. It is often assumed while formulating the problem that the controller has a sufficiently large bandwidth to track the commands from the guidance subsystem (Sutton *et al.*, 2000), however, in practice, true vehicle capability can only be measured in the presence of constraints such as system dynamics and actuator limitations.

The definitions and elements of a guidance loop discussed above are quite generic and refer to all guidance mechanisms. Although widely employed in the aerospace and land vehicles, it is equally valid for underwater vehicles.

2.2 Missile Guidance

The guidance technology of missiles is a mature field with an abundance of guidance laws already implemented in real systems. Many different guidance laws have been employed exploiting various design concepts over the years. Currently, the popular terminal guidance laws involve line of sight (LOS) guidance, LOS rate guidance, command-to-line-of-sight guidance, proportional navigation guidance (PNG)

(Locke, 1955), augmented PNG (APNG) (Zarchan, 1994) and optimal guidance law based on linear quadratic regulator theory, linear quadratic Gaussian theory or linear quadratic exponential Gaussian theory (Lin and Su, 2000). Among the current techniques, guidance commands proportional to the LOS angle rate are generally used by most high-speed missiles today to correct the missile course in the guidance loop. Recently, many advanced strategies have been implemented to generate different guidance laws. For instance, Rajasekhar and Sreenatha (2000) employed fuzzy logic to implement PNG law. The fuzzy law generates acceleration commands for the missile using closing velocity and LOS rate as input variables. The input data is fuzzified and their degree of membership to the output fuzzy sets is evaluated which is then defuzzified to get the acceleration command. A fuzzy based guidance law for missiles has also been proposed by Creaser *et al.* (1998) using an evolutionary computing based approach. The proposed law uses a genetic algorithm to generate a set of rules for the missile guidance law. Menon and Iragavarapu (1998) uses fuzzy logic weightings to blend three well-known guidance laws to obtain enhanced homing performance. The composite law evaluates the weights on each of the guidance systems to obtain a blended guidance command for the missile. Yang and Chen (2001) have implemented an H_∞ based guidance strategy. Unlike other guidance laws, it does not require the information of target acceleration, while ensuring acceptable interceptive performance for an arbitrary target with finite acceleration.

2.2.1 LOS guidance

LOS is the most widely used guidance strategy to date. In fact, almost all guidance laws in use today have some form of LOS guidance because of its simplicity and ease of implementation. The LOS guidance employs the line of sight angle λ between the vehicle and the target which can easily be evaluated using Equation 2.1.

$$\lambda = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{2.1}$$

where (x_1, y_1), (x_2, y_2) are the missile and target position coordinates respectively. The objective of the guidance system is to constrain the missile to lie as nearly as possible on the LOS. Since the missile ideally always lies on the line joining it to the target, the flight path will be a curved one. LOS guidance does not work well with manoeuvring targets. Moreover, the interception time is high which can be abridged using different strategies as discussed in the following sections.

2.2.2 PNG and its variants

The Lark missile that was tested in 1950 was the first missile to use PNG. Since then the PNG law has been used virtually in all of the world's tactical radar, infra-red and TV guided missiles (Rajasekhar and Sreenatha, 2000). It is the most common and effective methodology in case of non-manoeuvring targets that seeks to nullify the angular velocity of the LOS angle. In this technique, the missile heading rate is made proportional to the LOS rate from the missile to the target. The rotation of the LOS is measured by a sensor (either onboard or from a ground station), which causes commands to be generated to turn the missile in the direction of the target. Mathematically, the PNG law can be stated as

$$\eta_c = N' V_c \dot{\lambda} \tag{2.2}$$

where η_c is the acceleration command, N' is the navigation ratio, V_c is the closing velocity and $\dot{\lambda}$ is the LOS angle rate. The advantage of using PNG over LOS guidance is that the interception time can be greatly reduced by adjusting the navigation constant as shown in Figure 2.1 for the case of $N' = 1$ and $N' = 4$. In the latter case, the missile steering commands are four times as great. As a result the missile veers off much more to the left resulting in engagement.

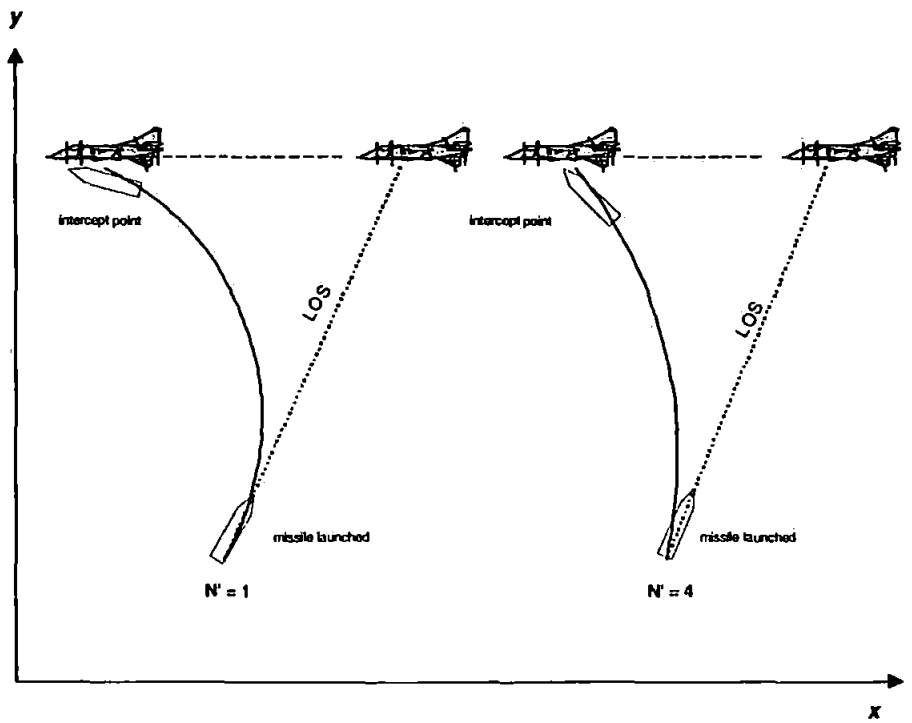


Figure 2.1: Proportional navigation guidance for a missile system

PNG like LOS guidance does not work well in the case of non-manoeuvring targets. However, the interception time is reduced. APNG is a modified form of PNG to deal with target manoeuvres.

Other forms of PNG are velocity compensated PNG, pursuit plus PNG, and dynamic lead guidance (Lin, 1991).

2.2.3 Optimal guidance law

Recently, great interest has been shown in using optimal control theory in the missile guidance problem. The two important mission parameters i.e. missile target engagement time and the energy needed can be reduced by utilising optimal control. Tsao and Lin (2000) proposed an optimal guidance law for short-range homing missiles to intercept highly manoeuvrable targets. The guidance problem that needs to be solved for the interception is to find the optimal missile trajectory such that the total time for the interception is minimised. The performance index J used in the proposed optimal law is

$$J = t_f = \int_0^{t_f} dt \tag{2.3}$$

where t_f is the interception time.

The proposed guidance law achieves the best performance in terms of the miss distance and interception time in comparison to the true PNG and APNG. However, a major disadvantage of this law is that the target's future trajectory must be known in advance which is impossible to evaluate in a realistic environment (Tsao and Lin, 2000). A comprehensive review of optimal guidance laws is documented by Lin (1991).

2.3 Guidance Laws for AUVs

The classical autopilots for AUVs are designed by controlling the heading or course angle in the control loop. By including an additional loop in the control system with position feedback from the sensors, an AUV guidance system can be designed. The guidance system generates reference trajectories to be followed by the vehicle utilising the data obtained by the navigation system.

The following section presents some important guidance laws found in the literature

which have been proposed or implemented in UUVs and are enlisted below.

- Waypoint guidance by LOS
- Vision based guidance
- Lyapunov based guidance
- Guidance using chemical signals
- Proportional navigation guidance for AUVs
- Guidance using magnetometers for cable tracking
- Electromagnetic guidance
- Guidance using long baseline and short baseline
- Fuel optimal guidance

2.3.1 Waypoint guidance by LOS

Waypoint guidance is the most widely used scheme in the field of AUVs. In the key paper by Healey and Lienard (1993), guidance is achieved between two points $[x_d(t_o), y_d(t_o)]$ and $[x_d(t_f), y_d(t_f)]$ by splitting the path between them into a number of waypoints $[x_d(k), y_d(k)]$ for $k = 1, 2, \cdots, N$ as shown in Figure 2.2. It is assumed that the vehicle is moving forward with speed U, then the LOS in terms of desired heading angle λ can be defined as

$$\lambda = \tan^{-1}\left[\frac{y_d(k) - y(t)}{x_d(k) - x(t)}\right] \tag{2.4}$$

where $[x(t), y(t)]$ is the current location of the vehicle. Care must be exercised to ensure that the heading angle λ is in the proper quadrant. When the vehicle reaches in the vicinity of the waypoint which is determined by checking if the AUV lies within a circle of acceptance (COA) of radius ρ_o around the waypoint $[x_d(k), y_d(k)]$ and if the vehicle's current location $[x(t), y(t)]$ satisfies

$$\rho^2 = [x_d(k) - x(t)]^2 + [y_d(k) - y(t)]^2 \leq \rho_o^2 \tag{2.5}$$

the next waypoint $[x_d(k+1), y_d(k+1)]$ is selected. The circle of acceptance could be taken as two times the length of the vehicle (Healey and Lienard, 1993).

If on the other hand, $d\rho/dt$ goes from negative to positive without the above condition

being met, than the waypoint has not reached. At this point, the guidance law must decide whether to keep the same destination waypoint and directing the vehicle to its COA or choose the next depending on a mission planning decision. A major disadvantage of the waypoint guidance is that, undesirable control energy consumption due to overshoot can be made during the change of trajectory. Therefore, selection of the reference trajectory for tracking is important to reduce the overshoot width of path and thus to decrease the control energy consumption. Yeo (1999) employed turning simulation to determine modified waypoints to avoid overshoot. Aguiar and Pascoal (1997) and Aguiar *et al.* (1998) proposed a modification in the waypoint guidance to deal with the presence of ocean currents. A current compensation for the heading autopilot has been developed which aligns the total vehicle velocity direction with the heading command.

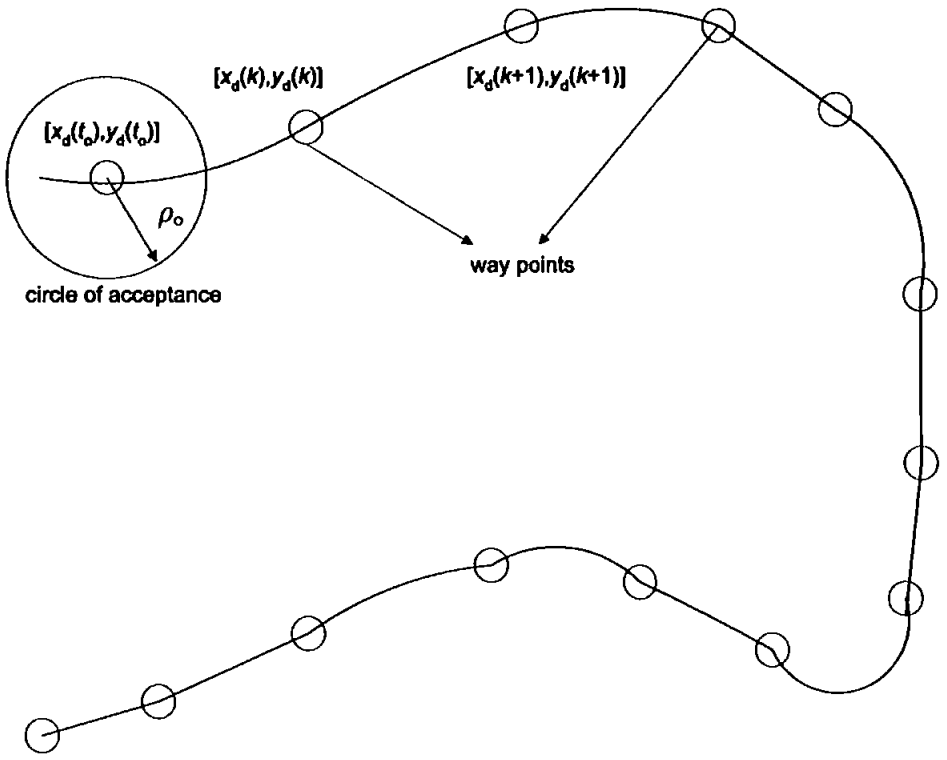


Figure 2.2: WayPoint guidance by LOS

Another LOS guidance scheme has recently been devised by Bakaric *et al.* (2004). The authors have provided an improved waypoint guidance scheme by adding a correction angle term to the desired heading of the vehicle to the next waypoint. This scheme aims at providing a smooth trajectory when transiting from one waypoint to another. This is depicted in Figure 2.3 where the desired heading is represented by ψ_{r0} and

correction is given by ψ_c. The correction angle is determined by incorporating the knowledge of the vehicle position, current waypoint, P_i and the next waypoint, P_{i+1}. The term ρ_0 represents the COA as defined before. The algorithm also allows for the compensation of sea current influence on the vehicle's motion, however, this is applicable when the current intensity is less than the vehicle's velocity.

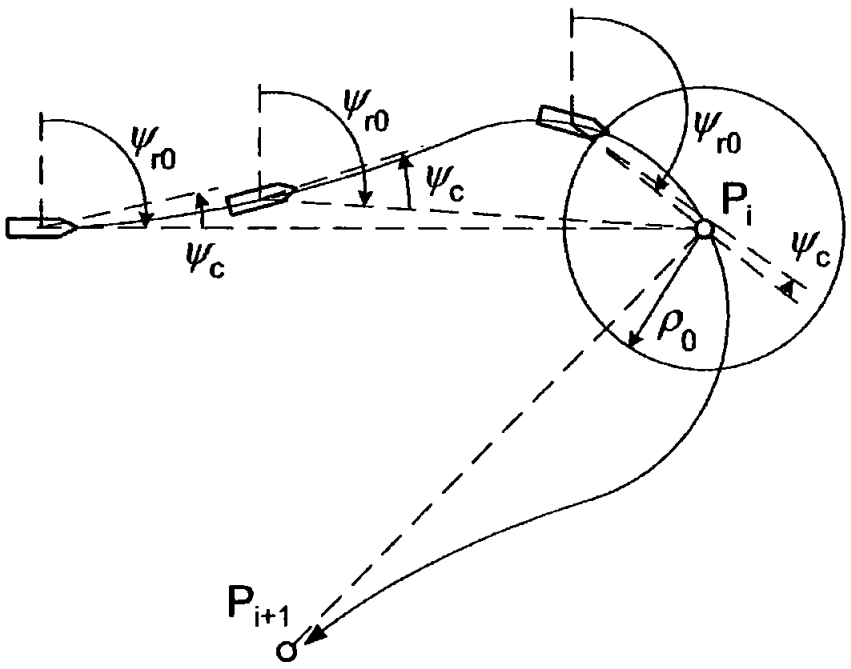


Figure 2.3: LOS guidance with reference heading correction (Bakaric *et al.*, 2004)

2.3.2 Vision based guidance

The vision based guidance technique has been inspired from the work of ROV operators, which utilise or rely on the visual information to perform tasks thus making a strong argument that visual imagery could be used to guide an underwater vehicle.

Vision based guidance has been mainly employed for cable tracking and docking problems (Wettergreen *et al.*, 1999; Balasuriya and Ura, 1998; Briest *et al.*, 1997; Rock *et al.*, 1992). Balasuriya and Ura (1998) developed an optical terminal guidance scheme and suggested its use for the docking of an AUV using a beacon. The beacon could be a light emitting device, which can be identified using photo detectors onboard the AUV. This scheme is analogous to a heat seeking air-to-air missile when locked on to its target. The disadvantage of using a beacon is that in shallow waters especially during the daylight, the photo detectors could be locked on to the sunlight.

A potential remedy is to adjust the frequency of the light emitted by the beacon.

Wettergreen *et al.* (1999) proposed vision based guidance for an AUV named *Kambara* using two cameras. The authors demonstrated that guidance can be achieved by a feature tracker algorithm which requires two correlation operations within the feature tracker as shown in Figure 2.4. The feature motion tracker follows each feature between previous and current images from a single camera whilst the feature range estimator correlates between the left and right camera images. The feature motion tracker correlates stored feature templates to determine the image location and thus direction to each feature. Range is determined by correlating the features in both images to find their pixel disparity. This pixel disparity is related to an absolute range using camera extrinsic and intrinsic parameters, which are determined by calibration. The direction and range to each feature is then fed to the controller which determines a set of thruster commands. To guide the AUV, thruster commands become a function of the position of visual features. A major drawback of using visual systems in underwater guidance is that the performance degrades in case of turbid water or when a cable is burried or there might be other similar cables appearing in the image. For such cases, a multisensor fusion technique has been proposed (Balasuriya and Ura, 1999*a*; Balasuriya and Ura, 1999*b*; Balasuriya and Ura, 2000; Balasuriya and Ura, 2001). The proposed sensor fusion technique uses deadreckoning position uncertainty with a 2D-position model of the cable to predict the region of interest in the image captured by a camera mounted on an AUV. The 2D position model of the layout of the cable is generated by taking the position coordinates (x_i, y_i) of a few points along the cable which is then used to predict the most likely region of the cable in the image.

As opposed to two cameras, Balasuriya and Ura (1998) proposed a vision based guidance law using a single camera. The technique has been implemented in a test bed underwater robot, *Twin-Burger 2*, at the University of Tokyo for cable tracking and a moving object. The basic idea underlying this scheme is that, the feature to be tracked introduces a particular geometric feature in the image captured by the CCD camera. The vision processor than label these features, extract their location in the image and interpret the appearance into a guidance parameter as shown in Figure 2.5. For example, an underwater cable introduces a line feature in the image and the edges of a cylinder introduce a rectangle. The vision processor derives the equation of the line representing the cable in the image plane given by Equation 2.6, which

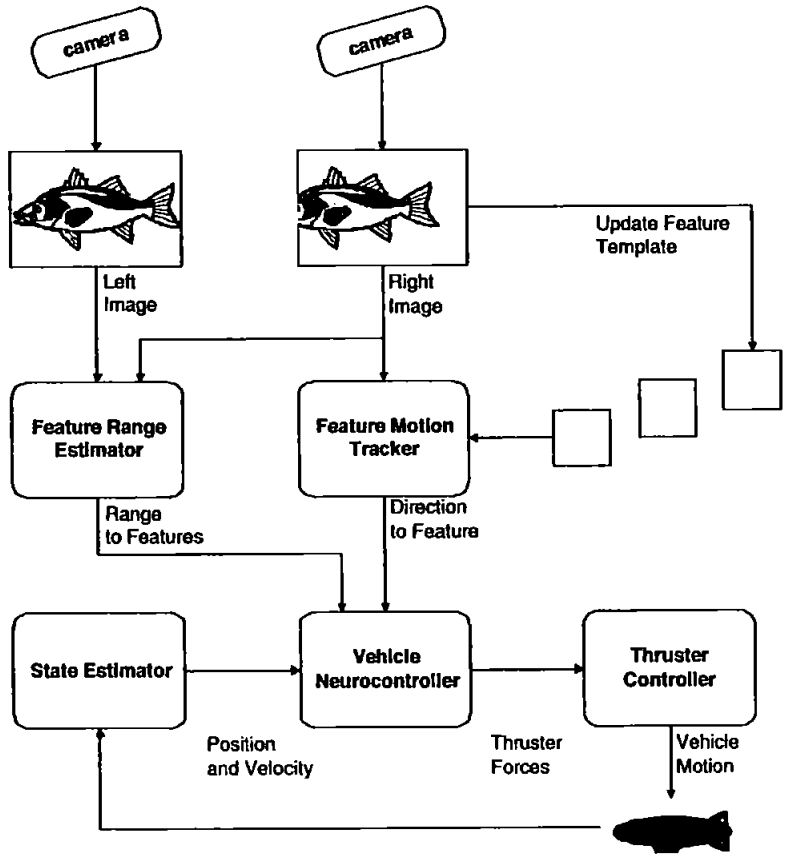


Figure 2.4: Vision based guidance System for the *Kambara* AUV

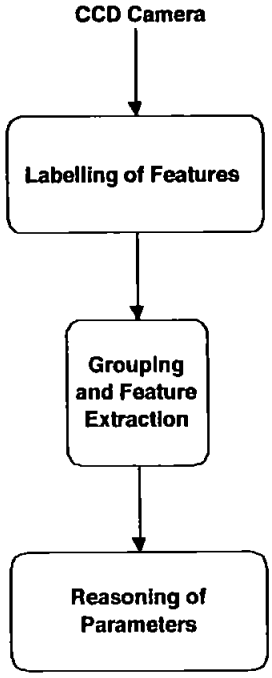


Figure 2.5: Vision based guidance System for the *Twin-Burger 2* AUV

gives the direction q and position r parameters.

$$r = x\cos(q) + y\sin(q) \tag{2.6}$$

where (x, y) are the coordinates of the straight line equation. In the case of a cylindrical object, the coordinates of the centroid of the object (rectangle) in the image plane and the area covered by the object are derived. These parameters are then fused with other sensory parameters to determine the control references for the underwater vehicle.

Rock *et al.* (1992) devised a vision based system to track a dot of light generated by a laser. The hardware is comprised of two cameras, one of which is used to locate the target. The vision system works by scanning the image from the last known location of the target, or from the centre of the screen if the target was not previously in view. The pixels are examined row by row, expanding outward towards the edge. If a target is found, its angle and elevation with respect to the centre of the image is evaluated and transmitted to the vision processor, whereas the range is determined using successive images from both cameras. The proposed law has been proved to be valid only in the case of a single distinguishable target.

2.3.3 Lyapunov based guidance

A Lyapunov function can be considered as a generalisation of the concept of distance or energy. The Lyapunov theorem states that if the distance of the state along any trajectory of $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$ decreases with time, then $\mathbf{x}(t)$ must tend to 0 as $t \rightarrow \infty$ (Chen, 1984). Caccia *et al.* (2000) uses the concept to develop a new guidance law for UUVs and tested on a prototype ROV called *Romeo*. This law is termed as the medium range manoeuvring guidance law. In this law, the vehicle is allowed to move from point (x, y) to (x_d, y_d) with a desired orientation ψ_d as shown in Figure 2.6. By choosing the desired vehicle speeds

$$u_d = \zeta e \cos\alpha \tag{2.7}$$

$$v_d = 0 \tag{2.8}$$

$$r_d = \mu\alpha + \zeta\frac{\cos\alpha\sin\alpha}{\alpha}(\alpha + h\theta) \tag{2.9}$$

where ζ, μ, and h are the tuning parameters, a Lyapunov function is suggested given by Equation 2.10, which makes the distance e between the two points converges to

zero for increasing time

$$V = \frac{1}{2}e^2 + \frac{1}{2}(\alpha^2 + h\theta^2) \tag{2.10}$$

where

$$e = \sqrt{(x_d - x)^2 + (y_d - y)^2} \tag{2.11}$$

$$\theta = \gamma - \psi_d \tag{2.12}$$

$$\alpha = \gamma - \psi \tag{2.13}$$

u_d, v_d and r_d are the desired vehicle's surge, sway and yaw velocities respectively. If an obstacle is detected along the way by the sensors with some orientation δ and range d from the robot, the vehicle follows its profile until a suitable detaching condition is verified and the vehicle then continues its free space manoeuvring. For feature following, the proposed law does not require the control of the vehicle sway velocity while controlling the surge and yaw velocities.

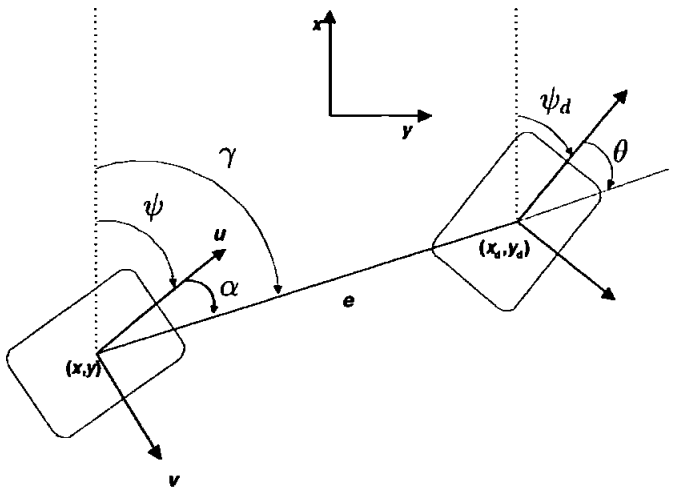


Figure 2.6: Medium range manoeuvring guidance law

2.3.4 Guidance with chemical signals

Using the fact that marine animals make extensive use of underwater chemical signals to avoid predators and to locate food sources etc., an interesting guidance scheme for AUVs using chemical signals has been proposed by Consi *et al.* (1994). The authors have built a small underwater robot, which mimics the chemical sensing abilities of a lobster. This class of robots is named as *biomimics*, which are designed to mimic certain features of animals and act as animal substitutes in behavioral and

neurobiological studies.

The goal of the paper is to use the information in chemical signals to locate the source of a chemical discharge. In this respect, it has a number of scientific, environmental, commercial and defence related applications. The sensors used in the *biomimic* are the conductivity sensors which are used to enable the AUV to follow a plume of saltwater in a freshwater flow-through flume. Simple gradient following algorithm is implemented to locate the source of discharge, which has the obvious disadvantage of getting trapped in local concentration minima and maxima.

Grasso (2001) documented a review on various chemoreceptors used by invertebrates to track the source of discharge for foraging or hunting purpose. A comparison is made between artificial sensors and sensors used by invertebrates under similar environmental conditions. A *biomimetic* robot was designed based on a Lobster's behaviour of chemical sensing and tracking. The robot receives guidance commands from an onboard gyro that provides a frame of reference to simulate estimation of flow direction. Three chemical detectors provide indications of plume extent in the boundary and viscous sublayers. The left and right sensors are used to detect the plume whilst the bottom sensor pointed towards the flume floor is intended to partially fulfill the role of the sensors on the legs of real lobsters.

2.3.5 Proportional navigation guidance for AUVs

Although PNG is widely used for missile guidance systems but Ahmad *et al.* (2003) demonstrated that it can be tailored to work for AUVs as well. The authors proposed a two-stage problem formulation to retrieve a returning AUV to the mother submarine. In the first stage, interception of the target (mother submarine) by the AUV is considered using PNG law, which is the theme of the paper. In the second stage, the docking of the AUV is considered when in close proximity to the mother submarine and is an area of current investigation. The idea behind using PNG is that if the AUV is made to lie on the LOS and hold it there as well, a constant relative bearing between the AUV and target is ensured i.e., the LOS does not rotate and interception will occur. The PNG law can be stated as

$$u_c \propto \lambda \tag{2.14}$$

$$u_c = k\lambda \tag{2.15}$$

where k is the navigation constant, λ is the LOS angle and u_c is the command input, therefore

$$u_c = NV_c\lambda \tag{2.16}$$

where V_c is the closing velocity and N is an important tuning parameter. The block diagram of the proposed guidance and control system is shown in Figure 2.7 where the guidance system used is PNG, which generate commands for the control system. Different engagement scenarios have been considered in this paper. For stationary targets, the scheme is analogous to the waypoint guidance. For mobile targets, the PNG law generates suitable trajectories to be followed by the AUV for docking purposes.

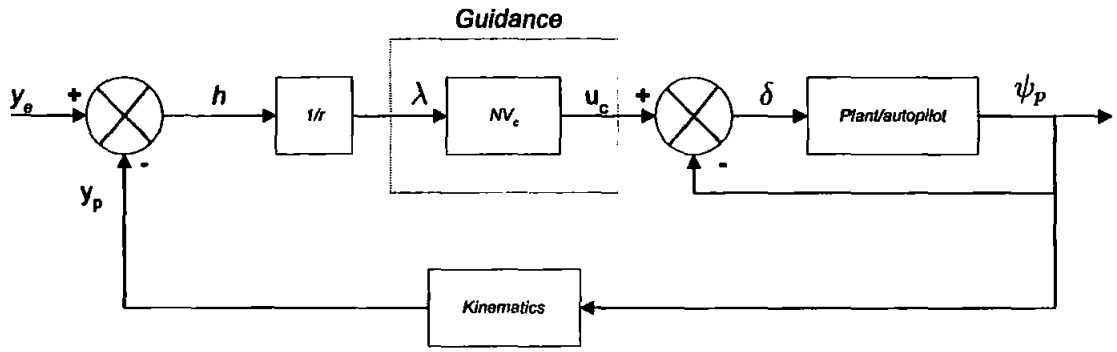


Figure 2.7: Proportional navigation guidance loop

2.3.6 Guidance using magnetometers for cable tracking

The underwater cable network and its capacity are expanding very rapidly, and its installation and maintenance becoming more important. AUVs could be a potential tool for underwater cable tracking for inspection purpose especially in case of deep waters where human intervention is not possible. Different schemes have been proposed for underwater cable tracking. Balasuriya and Ura (1998), proposed vision based guidance for cable/pipeline tracking as outlined in Section 2.3.2, but in the case of shallow waters, where cables are buried to avoid being damaged by fishing gears or anchors, the performance degrades. For buried electrical or telecommunications cables, the remedy is to use onboard magnetometers, which can detect the magnetic field induced from the current flowing in the cable. The data from the magnetometer is fed to a cable locator which estimates the direction, burial depth and the distance of the vehicle from the cable. The data from the cable locator is then used to guide the vehicle. The *Aqua Explorer 1000* is an example of a successful implementation of magnetometer based guidance for underwater cable tracking (Asakawa *et al.*, 1996; Ito

et al., 1994*a*; Kato *et al.*, 1994*a*). Guidance using magnetometers has limited applications as it can only be used to guide the vehicle towards the source of the magnetic field.

2.3.7 Electromagnetic guidance for AUV docking

A major disadvantage of using optical or visual guidance systems is that the response is only good in nonturbid, clear environments and it is limited over a wide range of background lighting and water turbidity conditions. In addition, the AUV must lie within the field of light emitted by the beacon on the cable or dock and should be oriented in such a way so that the optical sensors can detect the light. Feezor *et al.* (2001) employed an electromagnetic guidance (EM) technique during the homing/docking mode of an AUV. The EM guidance system uses a magnetic field generated by the coils on the dock which is sensed by the coils in the AUV. The guidance system provides to the AUV not only the bearing to the dock, but also the angle of the AUV relative to the field lines and thus the angle relative to the dock entrance. The accuracy of the proposed system is less than $20cm$ but the range is limited to $25 - 30m$. The proposed system is shown to be quite robust under almost all oceanographic phenomena.

2.3.8 Guidance using long baseline and short baseline

This guidance scheme is classified into two parts, the homing phase and the docking phase (Oh *et al.*, 2002). In the homing phase, a long baseline type sensor is used to estimate the position information whereas short baseline type sensor and a CCD camera is used during the docking part of the problem. Homing requires the path to the launcher to be generated by a cubic spline function which utilises the information related to the radius of curvature and path length. Assuming that the launcher is not stationary, an accurate path is derived by anticipating the position of the launcher (docking point). This is achieved by obtaining the information about velocities and positions of the AUV and the launcher and then generate an exact path by iteration. Since the launcher is continuously changing its position due to sea currents, therefore the path needs to be updated based on the new launcher position.

2.3.9 Fuel optimal guidance

A major problem all AUVs suffer is of limited onboard power. Most AUVs run off onboard lead acid batteries or fuel cells that have fixed capacities and therefore the range of mission gets confined. Kim and Ura (2003) addressed this problem by designing a fuel optimal guidance and tracking controller of AUV under current interaction. The total amount of fuel consumption to reach the terminal position is considered as the performance index to be minimised. The algorithm has been applied to the navigation of an AUV named *R-One Robot* developed at the University of Tokyo. The results illustrate apparent savings of fuel consumption, when the local distribution of sea current becomes more complicated and irregular.

2.4 Concluding Remarks

This chapter presents several guidance laws for unmanned vehicles with emphasis on UUVs. Guidance laws for airborne missile systems are also explored. It has been shown that the guidance system plays the vital role in bringing autonomy to the whole system. Moreover, it is observed that most of the current AUV systems employ either the classical waypoint guidance to reach a target area or the more advanced vision based guidance for cable/pipeline or object tracking. As a matter of fact, the LOS guidance is the key element of all guidance systems. Electromagnetic guidance was shown to have advantages over vision based methods but the applications are only limited to places where magnetic field is present. The discussion presented in this chapter provoked the formulation of two guidance laws in Chapter 5 for cruising type vehicles similar to *Hammerhead* for cable tracking missions. The first strategy suggested is another form of PNG which is termed as the pure pursuit guidance. The other system is a hybrid law combining several guidance schemes in different phases of the mission. The next chapter elaborates on *Hammerhead* hardware, its onboard sensors and the actuater control system followed by some trial results for system identification to obtain vehicle dynamics for controller design in Chapter 4.

Chapter 3

Hammerhead and Its Hardware Setup

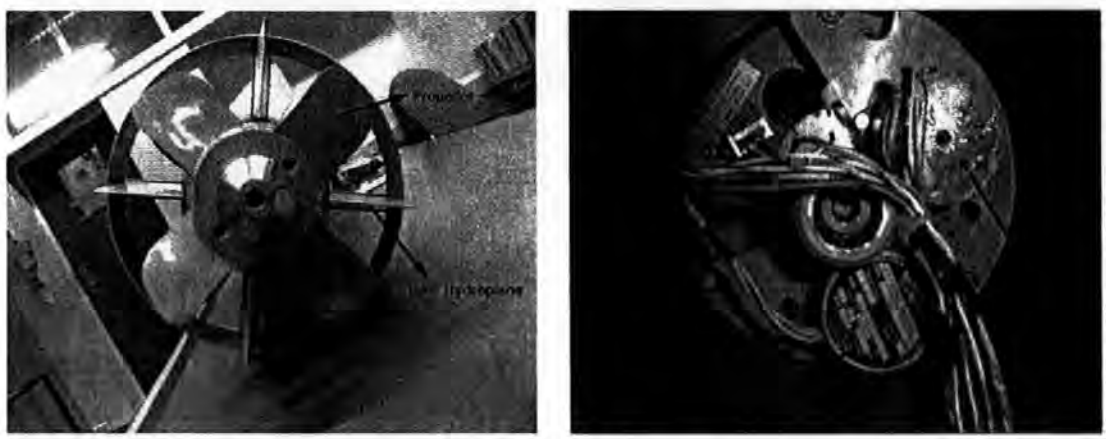
In this chapter, details concerning the hardware of *Hammerhead* is presented. It is shown how the *Hammerhead* configuration has evolved from an ROV type to fully autonoumous. The hardware developed at Plymouth and accompanying software is also reviewed. In particular, a description of the actuator controller card designed indigenously at Plymouth, the proximity sensors and a test rig to calibrate the programs for real time is explored along with the *Hammerhead* navigational suite.

3.1 Deep Mobile Target (DMT5)

Before being commissioned as an AUV, the *Hammerhead* used to be a deep mobile target (DMT) operated by the Royal Navy to simulate a submarine in training exercises. The vehicle was developed during the 1960s and acquired in the 1990s by the Offshore Technology Centre, Cranfield University. Since then, a number of modifications have been made to its original design and some new sections added to accommodate various sensors. In particular, the out of date electronics were stripped off and replaced by some state of the art sensors and actuators. The number 5 has been retained on the hull as shown in Figure 1.2 for historical purposes.

Dimensionally, the vehicle at present is approximately three and a half metres long and about one-third of a metre in diameter. The hull is made up of aluminum and divided into several sections with conventional 'O' ring seals which allows easy breakdown of the vehicle and the addition of extra modules. A counter-rotating electric motor drives the two propellers shown in Figure 3.1(a) at the rear end of the

vehicle. Also depicted in Figure 3.1(a) are the pair of rudders and hydroplanes for horizontal and vertical motion respectively. In the original DMT, the drive to the control surfaces was clutched from the main propulsion train. Each control surface had two clutches and the necessary position transducer to give control surface angle (Tetlow, 2001). This system has now been replaced with conventional stepper motors shown in Figure 3.1(b) that allow easy digital control and ideally do not require any feedback. However, a position feedback mechanism was later added for resetting purposes and will be discussed in the forthcoming sections.



(a) Rear rudder and sternplanes along with the counter-rotating propellers

(b) The rear stepper motors inside the *Hammerhead* hull

Figure 3.1: Rear rudder and hydroplanes on the original DMT and the new stepper motors

3.1.1 Transformation to an AUV

The DMT5 has a pair of hydroplanes at the rear end of the vehicle for diving. During the initial wet trials of *Hammerhead*, it was observed that the vehicle could not dive using the existing hydroplanes. This was due to the fact that the DMT5 was originally designed in such a way that it needs to be launched from a specially built platform to get it into the water. Such an arrangement is shown in Figure 3.2. Once the vehicle is submerged the sternplanes are responsible to control the depth.

This motivates the use of front hydroplanes or bowplanes on the *Hammerhead*. The planes were installed in an additional section together with the camera adjacent to the nose of the vehicle as shown in Figures 3.3(a) and 3.3(b). This is one of the stimulus of the name *Hammerhead* as the bowplanes give the vehicle a *Hammerhead Shark* like feature.

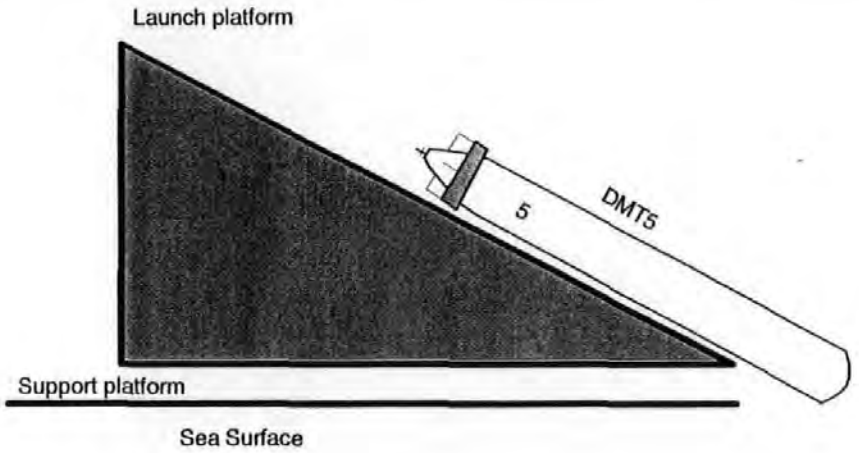
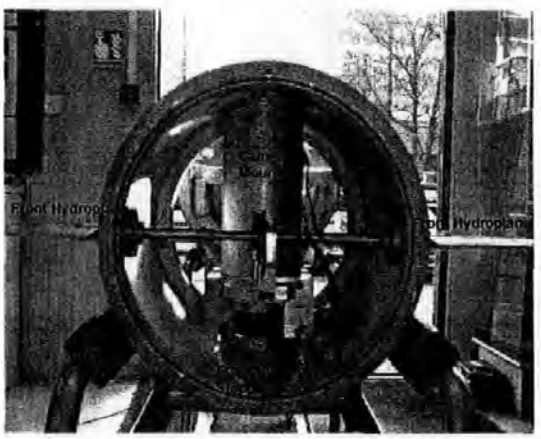


Figure 3.2: DMT5 being launched from a platform into the sea



(a) Physical layout of the front hydroplanes showing the stepper motor and camera section

(b) Front hydroplanes added to an extra section on *Hammerhead*

Figure 3.3: Front hydroplanes and the camera slot on an extra section next to the nose of *Hammerhead*

After careful trimming and repeated tank tests, the vehicle was kept slightly positively buoyant for ease of recovery. Test trials revealed that the vehicle was able to dive with the new hydroplanes. No physical constraints are imposed on the bowplanes, however, it is restricted to $\pm 25^0$ through software which was deemed adequate for diving and controlling depth and was obtained through a series of experiments. The sternplanes were secured and may be utilised in the future together with the bowplanes[1] for depth

[1]unless otherwise stated, the front hydroplanes or bowplanes are referred to by hydroplanes in general throughout the thesis

control.

The vehicle is powered from four onboard lead acid car batteries and are placed in the middle section as shown in the sectional view of *Hammerhead* in Figure 3.4. Also depicted are the sensor locations and an umbilical to communicate with the vehicle during preliminary testing. Since no power was being supplied externally, a very thin cable was used that could help minimise the drag effects. One end of the umbilical inside the vehicle was linked to all onboard sensors and to the actuator controller. Mission commands were being sent through the serial port of a laptop connected to the other end of the cable. This demonstrates an ROV-type configuration except that the power supply was internal.

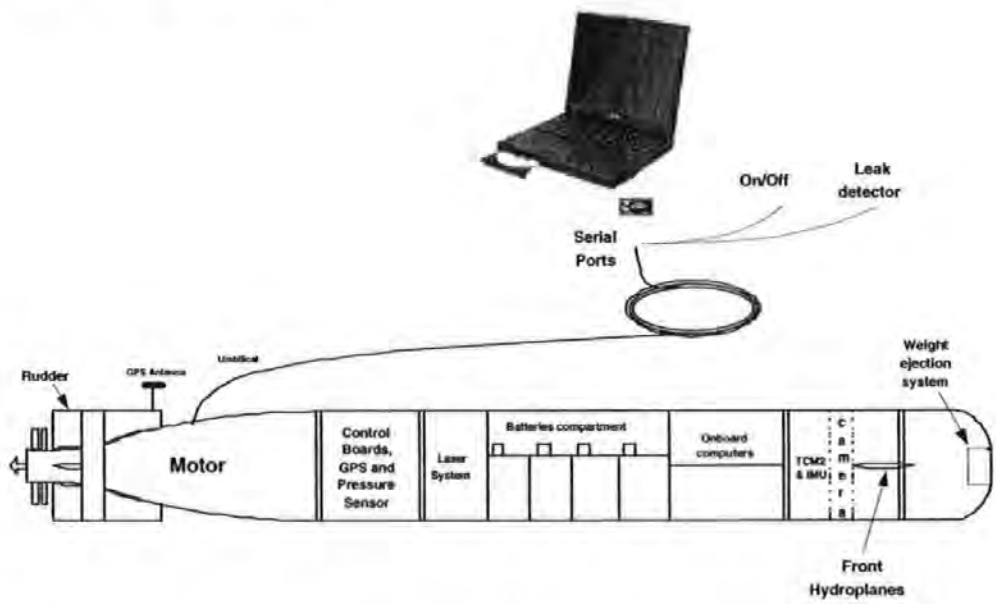


Figure 3.4: Sectional view of *Hammerhead* showing the umbilical and onboard sensor locations

The Matlab environment was used for all preliminary trials to obtain vehicle dynamics for system identification. However, the platform was subsequently modified to an integrated Matlab-Labview environment where Labview performs the data acquisition (DAQ) and vision system processing and Matlab computes the navigation and controller parameters. The *Hammerhead* is also equipped with two separate computers to deal with different software environments and the necessary DAQ hardware to interface with the sensors. The computers communicate with each other through a serial link at 9600 bits per second (bps). One serial port of the Matlab computer is used to query the Labview machine if there is any data string available whilst the

other port is directly communicating with the control surface actuators. It should be emphasised here that the *Hammerhead* is developed as an easy to use platform for the underwater research community for experimenting their NGC algorithms. The use of Matlab in this thesis is an example of the ease of developing the NGC software to be directly implemented on *Hammerhead* which offers an attractive feature to the underwater research groups where Matlab is generally used for system design and simulation.

It was noticed during the preliminary identification trials that the umbilical creates a severe amount of drag and the weight of the cable itself was affecting the vehicles movement. In addition, during another series of wet trials at Roadford Reservoir, Devon, the cable was chopped off by the propeller and the mission had to be aborted prematurely costing time and money. These factors influenced the choice of developing a wireless ethernet connection to communicate with the vehicle for all future missions. The front end of the communication software was designed and developed at Cranfield University in the Labview environment and is used for initialisation, relaying mission parameters, viewing the state of the vehicle through the camera in real time and to interrupt the mission in case of an unlikely event. In this way, the vehicle was executed autonomously with little human supervision. A number of precautions were implemented in order not to lose the vehicle in the early stages of its development during autonomous runs. The weight ejection system in the nose of the vehicle, also shown in Figure 3.4, is one of several precautions and is activated when the vehicle goes beyond a specified depth, thereby, increasing its buoyancy to recover it on the surface. Since the *Hammerhead* is kept slightly positively buoyant, turning off the main thruster could also help in recovery in case of an unwanted scenario.

The forthcoming sections detail the *Hammerhead* hardware, in particular, the actuator controller and the navigational sensors.

3.2 Control of Onboard Actuators

As mentioned above, the rear rudder and front hydroplanes are controlled by two onboard stepper motors. The major advantage of using stepper motors is that direct digital control is possible and it does not require any feedback mechanism besides being able to control its movement precisely. For this purpose, an interface was sought which could receive an input from the onboard control computer and converts it into

the stepper motor driving pulses. Several such type of interface units are available off-the-shelve, however, it was decided to design it indigenously at Plymouth for various reasons such as ease of modification, duplication in case of failure, custom made design to accommodate it in the vehicle etc. A microcontroller based board was thus developed as opposed to using the traditional stepper motor driving chips such as SAA1027. The microcontroller used is an Atmel 89C2051 with an on chip serial port which can communicate with the control computer thus providing a simple digital interface. The AT89C2051 has an 8 bit port which is programmed to drive the stepper motors. Auxilliary transistors are added in between to increase the driving current to the motors. The top 4 bits of the port are connected to the hydroplanes whereas the lower part generate the driving pulses for the rudder actuator. The microcontroller receives one byte of information from the control computer and translates it into necessary driving pulses for the rudder or hydroplanes. The assembly code and schematic are provided in Appendix A and the information code is illustrated in Figure 3.5.

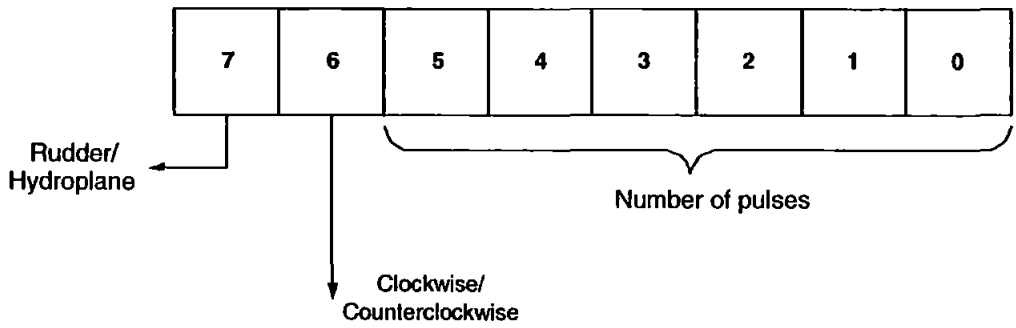


Figure 3.5: Information byte send by the control computer to the actuator interface board

As shown, the most significant bit 7 is used to identify if the command is for the rudder or hydroplane where a '1' indicates rudder and a '0' represents hydroplane. Bit 6 is for the direction i.e. clockwise or counterclockwise. This translates into appropriate command for rudder (LEFT or RIGHT) and hydroplane (UP or DOWN). The last 6 bits are the number of pulses such that for every received pulse, the motor turns its minimum step angle. A typical stepper motor driving sequence is shown in Figure 3.6. This gives the pulses represented by an **A** followed by $\mathbf{9}, \mathbf{5}$ and **6**. To change the direction, this sequence is simply reversed.

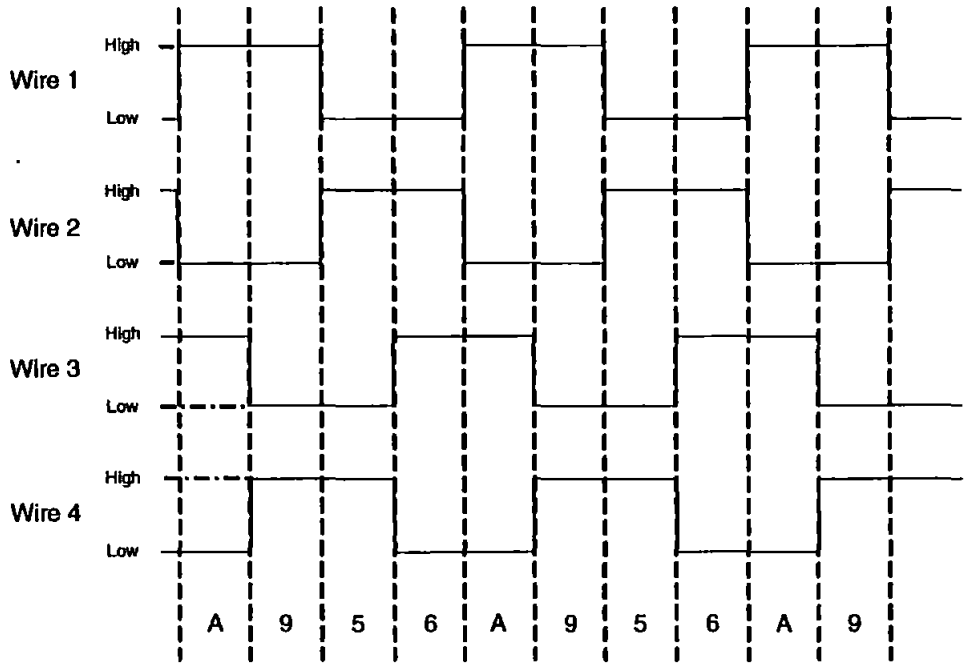


Figure 3.6: A typical stepper motor driving sequence

3.2.1 Proximity sensors

The proximity sensor can help provide the exact location of the control surfaces during any phase of the mission. These are used as feedback to the actuator controller to correct for the demanded position and to reset the control surfaces at the beginning of every mission. In addition, sometimes there is a need to abort the mission prematurely resulting in the control surfaces position being unknown. This costs a considerable amount of effort and time to reset the planes manually by observation. Two simple potentiometers were therefore installed each for the rudder and hydroplane actuator which provides position of the control surface in terms of voltage. This voltage is used as a feedback to correct the deflection angle and to re-initialise the control surfaces before commencing any mission.

A test rig was developed to debug the source code and to calibrate the mission parameters for real time. The test rig included a stepper motor and a controller assembled in a similar fashion as in the vehicle and is shown in Figure 3.7. A proximity sensor is also connected for position feedback which is visible and attached to one end of the shaft whose other end is tied to the control surface.

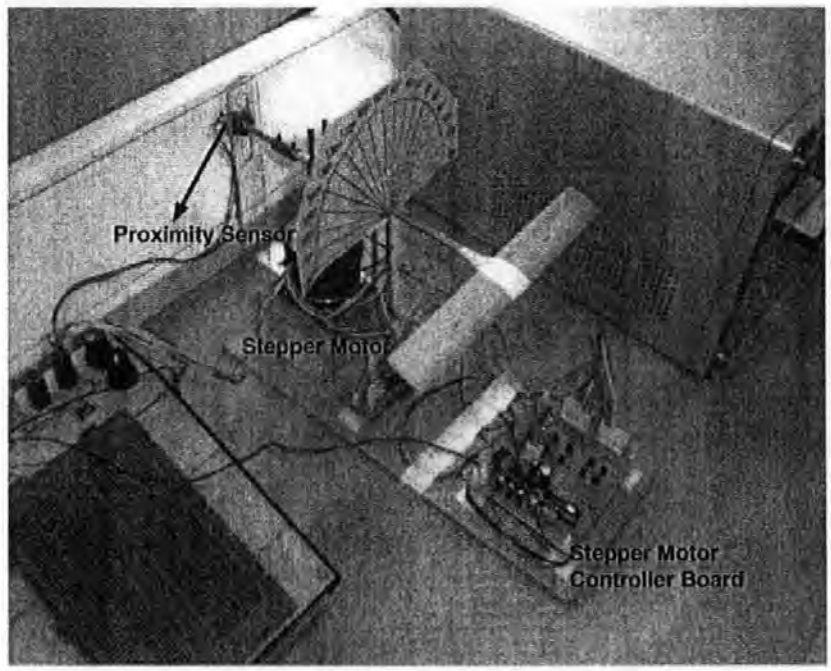


Figure 3.7: Test rig developed to calibrate the mission programs for in water experiments showing a stepper motor, control board and proximity sensor

3.3 *Hammerhead* Navigational Suite

Navigation is a key element in an NGC strategy that determines the location of the vehicle and target. A reliable navigation system is required which provides accurate information about the vehicle state and ideally must be error free. A fault in the navigation system will reflect directly on the control system performance.

Generally, a global positioning system (GPS) is employed to pinpoint any location on the surface whilst dead reckoning is the method of establishing the position underwater. Some other forms of acoustic navigation techniques to provide positional information are long base line (LBL), short base line (SBL), Doppler velocity log (DVL) to name but a few. Reader is referred to Loebis *et al.* (2002) for a comprehensive review of various AUV navigation techniques.

The manufacturing cost of an AUV largely depends on the sensor pay load on it and can be kept minimum by careful selection of reasonably accurate low cost sensors. The *Hammerhead* sensor suite was procured under these limitations since it was designated as a low cost AUV (see Chapter 1). The requirements of a simple heading and depth

control mission is a compass and a depth sensor. For waypoint following, the position information is also needed which can be estimated using an inertial measurement unit (IMU)(underwater and on surface) or a GPS (surface only). To track an object or for obstacle avoidance purpose, a sonar or vision system is usually employed. Clearly, the choice of an AUV navigation suite depends on the type of mission. It is also very common that a sensor gives erroneous measurements during a mission which can lead to an overall performance degradation. To eradicate this problem, redundant data is needed and an MSDF technique (Loebis *et al.*, 2002) is exploited to deal with faulty sensors.

At present, the *Hammerhead* is equipped with all basic navigation sensors such as a Garmin GPS to provide latitude and longitude, a precision navigation TCM2 electronically gimballed compass for the Euler angles (yaw, pitch and roll), an IMU by Watson Industries that gives linear accelerations, angular rates and Euler angles and a pressure transducer that provides the depth of the vehicle.

The latitude and longitude from a GPS receiver can be converted to world coordinates for positioning on the surface and dead reckoning can be performed using IMU measurements underwater. Since the IMU generally provides accelerations therefore it has to be double integrated to get the position information. Unfortunately, dead reckoning technique accumulates an error over time therefore a surface fix is needed from the GPS. A vision system is also installed on the vehicle which is very useful for tracking an object on the seabed and for positional and velocity information (Dalgleish *et al.*, 2003; Loebis *et al.*, 2003) when the seabed is in the field of view of the camera. In addition to all this, the *Hammerhead* is equipped with a *forward looking sonar* which could be an alternate and an invaluable source of information for objects underwater even if the water is not clear. Moreover, it serves as a viable tool for obstacle avoidance purpose and ocean surveying to create seabed maps. An obstacle detection and avoidance system is being developed at Plymouth and is an area of ongoing research (Tan *et al.*, 2003*a*; Tan *et al.*, 2003*b*; Tan *et al.*, 2004). Figure 3.8 shows the flow of the navigational information from the sensors to the navigation subsystem which is subsequently integrated with the guidance and control system.

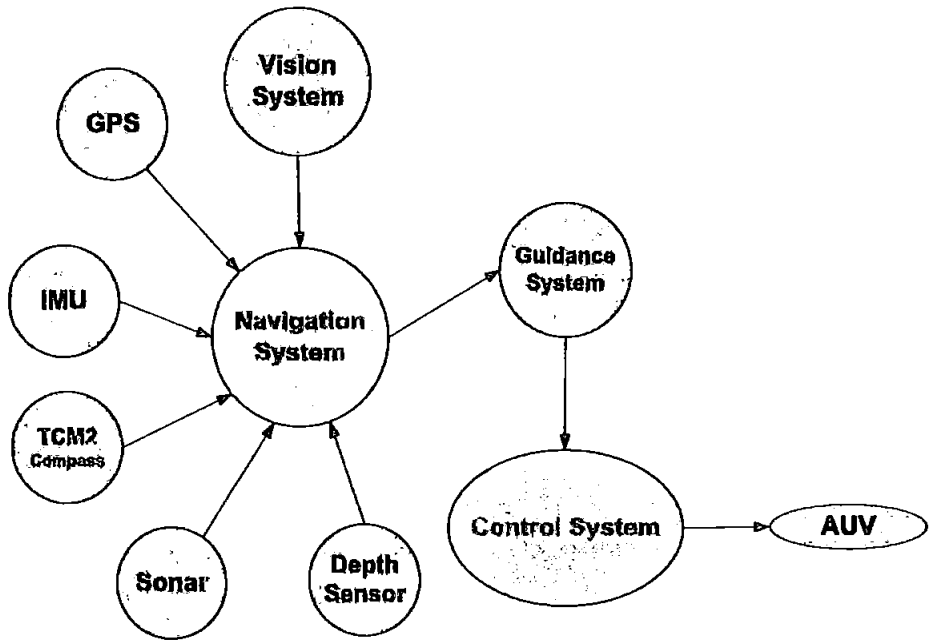


Figure 3.8: Flow diagram of a navigation, guidance and control system showing all the sensors on the *Hammerhead* AUV

3.3.1 Sensor strings

All sensors output data is in the form of strings. The strings can be read from the analogue output of the instrument or directly be imported into a computer through a serial interface. Some sensors provide additional data that is not essential for the navigation system, therefore the summary below provides only the main parameters of the strings.

TCM2 Electronic Compass

A typical TCM2 string is given by

$\$C102.1P2.1R10.8$

Here, \$ marks the beginning of string, C represents the heading angle with respect to the magnetic north in degrees and it ranges from 0 to 360^0. P provides the pitch angle in the limit $\pm 20^0$ while R is the roll angle in degrees and saturates at $\pm 20^0$.

Inertial Measurement Unit

The IMU can be configured to give several outputs including accelerations and Euler angles. A typical output from IMU may look like this

I +179.1 -01.0 250.0 -0.01 -0.01 +1.00 -0.00 $\cdots$

The I in the beginning is the initialisation of the string. The rest of the string is user programmable. Some of the several outputs which the *Hammerhead* IMU can provide are listed in Table 3.1 below.

Parameters
◇ Euler angles i.e. roll, pitch and yaw of the vehicle
◇ Accelerations along the x, y and z-axis in body coordinate frames
◇ Forward, lateral and vertical accelerations in earth coordinate frames
◇ Angular rate along the x, y and z-axis
◇ Heading angle rate
⋮

Table 3.1: IMU string parameters

Global Positioning System

Each output string of a GPS is called a *sentence* which can be configured using vendor software. The GPS *sentence* usually starts with a '$' sign followed by the *sentence* name such as 'GPGGA' or 'GPRMC' followed by a list of available data. An archetypal GPS *sentence* is shown below starting with a GPGGA

$GPGGA,130432,5204.3658,N,00037.6113,W,0,00,,,M,,M,, ,

The parameters required from the GPS are latitude and longitude which can be transformed into world coordinates, thereby, providing position information. The vehicle forward velocity is also available from the GPS receiver which can be fused with other velocity outputs from the IMU.

Pressure Transducer

The pressure transducer converts pressure exerted by water to an equivalent analogue voltage. This voltage can be fed directly to any analogue channel on a DAQ board or can be converted to digital signals using an analogue to digital converter (ADC), which can then be read through the serial port of a computer. This voltage is finally converted into equivalent pressure and thus depth of the vehicle is obtained.

3.4 Concluding Remarks

This chapter has provided a detailed description of the *Hammerhead* hardware. It has been shown how the *Hammerhead* configuration has changed, initially based on an umbilical, to a wireless communication based system. Details on the actuator controller card and associated hardware and software is also elaborated. Finally, the navigational sensors and the data available to the MSDF algorithm is reviewed. Chapter 4 provides additional and in depth knowledge on the working of *Hammerhead* hardware during the preliminary system identification experiments.

Chapter 4

System Identification and Modelling

This chapter aims at developing a model of the *Hammerhead* AUV for controller design. A brief description of the mathematical modelling technique is presented followed by system identification (SI) theory and its application to *Hammerhead*. Experimental data collected for SI is reviewed and results obtained by applying SI to the data are shown. The algorithm developed for data acquisition experiments is also explained. Separate models for the rudder-yaw and hydroplane-depth channels are estimated.

4.1 System Modelling

All controller designs are based on a model of the physical system to be controlled. This gives the modelling process utmost importance before any real time controller can be developed. It is imperative that the designer gain significant depth into system behaviour via extensive simulations using a model of the process as an alternative to the physical system. Clearly, this requires a model that can replicate the systems dynamic behaviour as closely as possible.

Modelling an underwater vehicle is a complex task because of the nonlinear nature of the vehicle dynamics and the degrees of freedom of vehicle movement. In addition, cross coupling effects make the controller design even more intricate. Fortunately, there is a plentiful amount of literature available on the mathematical modelling of underwater vehicles and it is generally applicable to all types of underwater vessels. However, a major difficulty in using these generalised models is the evaluation of

hydrodynamic coefficients which require tank tests on a full scale physical model of the vehicle provided the test facility is available. The Subzero-III vehicle based at The Institute for Sound and Vibration Research, University of Southampton, UK, for instance, has over 70 rigid body and hydrodynamic coefficients to be estimated. Twelve of these were obtained by calculation or experiments. Lack of tank test facilities reduce the evaluation of the coefficients to only four whereas the remaining coefficients used were scaled down versions of the Ocean Voyager vehicle, which is similar in shape to Subzero-III (Ahmad and Sutton, 2003).

An alternate route to modelling an AUV is thus suggested and used in this thesis using SI techniques on observed data obtained from test trials. More will be said about SI in the next section. In the following, some generic equations used in the mathematical modelling of an underwater vehicle are elaborated.

4.1.1 Mathematical modelling of an AUV

The generalised six degree of freedom (DOF) rigid body equations of motion of an AUV are given by Fossen (1994) as

$$\mathbf{M}_{RB}\dot{\nu} + \mathbf{C}_{RB}(\nu)\nu = \tau_{RB} \tag{4.1}$$

here $\nu = [u\ v\ w\ p\ q\ r]^T$ represents the linear and rotational motions of the rigid body in body-fixed coordinate system. $\mathbf{M}_{RB}$ is the rigid body inertia matrix satisfying

$$\mathbf{M}_{RB} = \mathbf{M}_{RB}^T > 0; \qquad \dot{\mathbf{M}}_{RB} = 0$$

and the matrix $\mathbf{C}_{RB}$ correspond to the Coriolis and centripetal forces that can be parameterised to a skew symmetric matrix i.e.

$$\mathbf{C}_{RB}(\nu) = -\mathbf{C}_{RB}^T(\nu)$$

$\tau_{RB} = [X\ Y\ Z\ K\ M\ N]^T$ is a generalised vector of external forces and moments about the origin acting as an input to the system. More precisely, τ_{RB} can be written as

$$\tau_{RB} = \tau_H + \tau_E + \tau$$

where τ_H includes the hydrodynamic forces and moments. τ_E describes the environmental forces and moments acting on the vehicle and τ is the propulsion forces and

moments. Hence Equation 4.1 can be written as

$$\mathbf{M}\dot{\nu} + \mathbf{C}(\nu)\nu + \mathbf{D}(\nu)\nu + g(\eta) = \tau \tag{4.2}$$

In this equation,

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_{RB} + \mathbf{M}_A \\ \mathbf{C}(\nu) &= \mathbf{C}_{RB}(\nu) + \mathbf{C}_A(\nu) \end{aligned}$$

here $\mathbf{M}_A$ is the added mass matrix, $\mathbf{C}_A$ is the Coriolis and centripetal effects due to added mass, $\mathbf{D}(\nu)$ is the hydrodynamic damping matrix, g is the vector of gravitational and buoyant generalised forces, τ represents the forces and moments acting on the vehicle in the body-fixed coordinate system and $\eta = [x\ y\ z\ \phi\ \theta\ \psi]$ is a vector of position and Euler angles in earth-fixed frame of reference. Expanding Equation 4.1 gives

$$m\left[\dot{u} - vr + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r} + z_G(pr + \dot{q}))\right] = X \tag{4.3}$$

$$m\left[\dot{v} - wp + ur - y_G(r^2 + p^2) + z_G(qr - \dot{p} + x_G(qp + \dot{r}))\right] = Y \tag{4.4}$$

$$m\left[\dot{w} - uq + vp - z_G(p^2 + q^2) + x_G(rp - \dot{q} + y_G(rq + \dot{p}))\right] = Z \tag{4.5}$$

$$\begin{aligned} I_x\dot{p} + (I_z - I_y)qr &- (\dot{r} + pq)I_{xz} + (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} \\ &+ m\left[y_G(\dot{w} - uq + vp) - z_G(\dot{v} - wp + ur)\right] \end{aligned} = K \tag{4.6}$$

$$\begin{aligned} I_y\dot{q} + (I_x - I_z)rp &- (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + (qp - \dot{r})I_{yz} \\ &+ m\left[z_G(\dot{u} - vr + wq) - x_G(\dot{w} - uq + vp)\right] \end{aligned} = M \tag{4.7}$$

$$\begin{aligned} I_z\dot{r} + (I_y - I_x)pq &- (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + (rq - \dot{p})I_{zx} \\ &+ m\left[x_G(\dot{v} - wp + ur) - y_G(\dot{u} - vr + wq)\right] \end{aligned} = N \tag{4.8}$$

(Please refer to the nomenclature in the beginning of the thesis for a description of notation used). The first three equations represent the translational motion while the last three equations represent the rotational motion of the AUV. Figure 4.1 depicts these quantities and the coordinate reference frames. The centre of gravity of the vehicle is assumed to be at (x_G, y_G, z_G). By coinciding the centre of gravity with the origin O, the above equations can be greatly simplified. It should also be noted that the velocity vector ν in body coordinate frame cannot be directly integrated to obtain the position coordinates in earth-fixed reference frame, rather, they are related by the transformation matrix $T(\eta)$ given by (please refer Kuipers (1999) for a derivation)

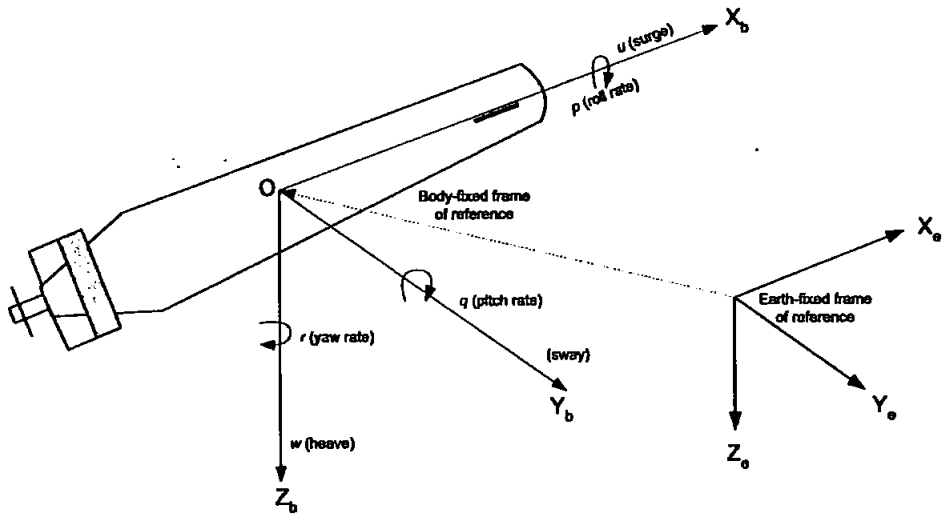


Figure 4.1: Earth-fixed and body-fixed reference frames

$$T(\eta) = \begin{bmatrix} T_1(\eta_2) & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times3} & T_1(\eta_2) \end{bmatrix}$$

where

$$T_1(\eta_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}$$

$$T_2(\eta_2) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix}$$

and

$$\eta_2 = [\phi \quad \theta \quad \psi]^T$$

In the above transformations, $s\cdot = \sin(\cdot)$, $c\cdot = \cos(\cdot)$, $t\cdot = \tan(\cdot)$ and $\mathbf{O}_{3\times3}$ is a null matrix. This gives the vector $\dot{\eta}$

$$\dot{\eta} = T(\eta)\nu \tag{4.9}$$

which can be integrated to get the position coordinates in earth-fixed frame of reference. Please note that the model obtained using this method is nonlinear which for controller design purposes, linearised to some operating point. However, the SI technique generally provides a linear model which is valid only for a given operational conditions. The next section discusses SI in the context of underwater vehicles followed by its application to *Hammerhead*.

4.2 System Identification

SI offers an alternate route to model an underwater vehicle in contrast to a painstaking mathematical modelling technique. This approach is quite useful in providing reliable and accurate models in a short time using only input/output data and without the need to estimate the hydrodynamic coefficients requiring tank tests. This feature, therefore, is attractive for the underwater vehicle manufacturers, where a vehicle configuration changes frequently to suit the mission requirements. Moreover, by experimenting with the vehicle to obtain the data, significant insight about the vehicle dynamics can be gained thereby assisting in model development. AUV modelling using SI approaches have been investigated before (Tinker *et al.*, 1979; Ippoliti *et al.*, 2001; Goheen and Jefferys, 1990; Ahmad and Sutton, 2003; Bossley *et al.*, 1999), but most of the work involved has been undertaken on identifying a model by generating data from a 6 DOF mathematical model of the vehicle. However, in this thesis, the SI is performed on actual AUV input/output data obtained from test trials and is explained in the next section.

The SI of a dynamical system generally consists of the following four steps

1. Data acquisition (DAQ)
2. Characterisation
3. Identification/estimation
4. Verification

The first and most important step is to acquire the input/output data of the system to be identified. Acquiring data is not trivial and can be very laborious and expensive. This involves careful planning of the inputs to be applied so that sufficient information about the system dynamics is obtained. If the inputs are not well designed, then it could lead to insufficient or even useless data. Other factors that could degrade the data quality includes the DAQ hardware involved and sampling rate. These will be discussed in detail in the following section.

The second step defines the structure of the system to be identified, for example, type and order of the differential equation relating the input to the output. This means selection of a suitable model structure, e.g. auto-regressive with exogeneous input (ARX), auto-regressive moving average with exogeneous input (ARMAX), output

error etc. If there is significant amount of noise in the data then it could be modelled separately by specifying an appropriate model type. A generic input-output linear model for a single output system can be written as (Ljung, 2001)

$$A(q^{-1})y(t) = \sum_{i=1}^{nu} \left[\frac{B_i(q^{-1})}{F_i(q^{-1})}\right] u_i(t - nk_i) + \left[\frac{C(q^{-1})}{D(q^{-1})}\right] e(t) \tag{4.10}$$

where u and y are the input and output respectively and u_i represents the ith input. A, B_i, C, D and F_i are polynomial in the delay operator q^{-1}, nk denotes the time delay in the system and e is the disturbance. All the above mentioned models can be obtained from the generic model structure by substituting the appropriate values of the polynomials.

The third step is identification/estimation, which involves determining the numerical values of the structural parameters, which minimise the error between the system to be identified, and its model. Common estimation methods are least squares (LS), instrumental-variable (IV), maximum-likelihood (MLE) and the prediction-error method (PEM). A common criterion used in most optimization methods is the quadratic error function given by

$$\min_{\Gamma} \; J = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}(k) - y)^2 \tag{4.11}$$

where $\hat{y}$ is the predicted output from the model, y represents the actual output, N denotes the number of data points and Γ contains the coefficients to be estimated in a given model structure.

The final step, verification, consists of relating the system to the identified model responses in time or frequency domain to instil confidence in the obtained model. Residual (correlation) analysis and cross-validation tests are generally employed for model validation. The residuals ε are defined as the difference between the model output and measured output. For a perfect model, the residuals should reduce to an uncorrelated sequence e with zero mean and finite variance. Correlation based tests are employed to verify if

$$e(t) = \varepsilon(t) \tag{4.12}$$

This is achieved by verifying if the correlation functions are within the confidence intervals i.e.

$$\phi_{\varepsilon\varepsilon}(\kappa) = E\left[\varepsilon(t-\kappa)\varepsilon(t)\right] = \delta(\kappa) \tag{4.13}$$

$$\phi_{u\varepsilon}(\kappa) = E\left[u(t-\kappa)\varepsilon(t)\right] = 0 \tag{4.14}$$

where $\phi_{\varepsilon\varepsilon}$ and $\phi_{u\varepsilon}$ represents the autocorrelation of residuals and cross correlation of residuals and input respectively. u is the excitation signal to the system and δ is the dirac delta function defined as

$$\delta(\kappa) = \begin{cases} 0 & \text{if } \kappa \neq 0 \\ 1 & \text{if } \kappa = 0 \end{cases}$$

If the cross correlation test in Equation 4.14 is not verified, this means that there is something in the residuals which is originating from the input and has not been properly taken care of by the model and thus the model needs further tuning.

The above-mentioned features of SI are symbolically indicated in Figure 4.2 where $d(t)$ is the external noise or disturbance to the plant. SI theory is well established and the reader is referred to Ljung (1999) for a comprehensive treatment.

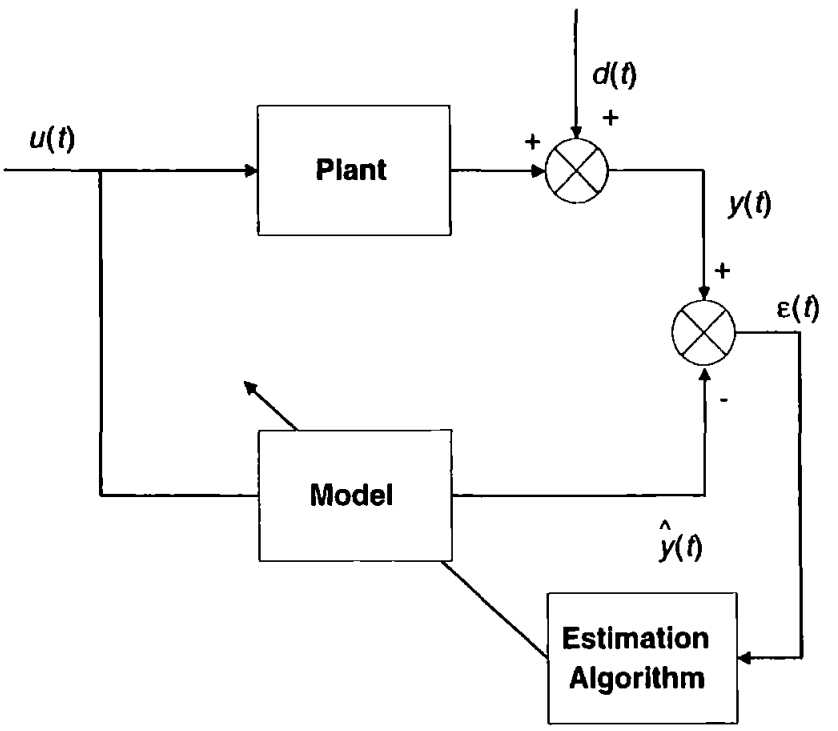


Figure 4.2: The overall system identification procedure

4.2.1 General considerations for SI trials

As explained in Chapter 3, for the preliminary identification trials, an umbilical was used to send and receive signals from the vehicle connected to a laptop computer on a pursuing boat. Although, the effect of the umbilical was quite severe, in particular, on the depth dynamics, it was decided to carry out the experiments with the existing setup. The cable was later removed and some additional tests were performed to improve the model quality. The cable drag was not modelled separately and is considered as an integral part of the vehicle dynamics. It will be shown in subsequent chapters that the initial in-water controller tests were performed using the same arrangement including the cable, thereby, not affecting the overall dynamics.

There were, however, a few considerations in designing the experiments for SI of *Hammerhead*, in particular, with regard to the input signal type, the sampling period and the length of experiments. Most of the rudder-yaw channel identification trials were performed just under the surface. In this case, therefore, ideally any type of input sequence can be designed to excite the system and can be extended to any duration as long as the vehicle is away from the shore. The hydroplane-depth channel was rather tricky and involved careful planning of the input signal and the duration of experiment. The length of experiments in this case had to be constrained because (a) the cable length was finite and (b) long duration missions would risk the vehicle hitting on the seabed or bottom of the reservoir without the depth autopilot in the loop resulting in catastrophy.

Another factor that influences the observed data quality is the sampling rate. This is system dependent and the designer should select it carefully based on his observation and experience before identifying a model. Too low sampling rate could result in little or incomplete information about the system dynamics. This is because important trends in the system output could be missed out due to undersampling. On the other hand, if the system is sampled too rapidly, a low signal-to-noise-ratio (SNR) results and the noise dynamics dominate the actual behaviour of the system, thus affecting the overall data quality. An alternate way to check the sampling rate is to analyse the pole-zero plot of the estimated model. If the poles and zeros are clustered tightly around the origin, the system has been sampled too slowly, however, if the poles and zeros are packed around the unit circle ($|z| = 1$), the system is sampled too fast. A reasonable spread of pole-zero plot locations is thus desired in the z-plane within the unit circle.

In case of *Hammerhead* , the sampling frequency need not necessarily be too high because of the slow dynamics of the vehicle. Too high sampling rate will not provide any additional information but could lead to more noise in the data which needs to be filtered out or modelled separately. The input frequency should also be constrained and should not be more than a specified value which can be obtained iteratively based on experience and observation. Higher input frequency ranges will provide no information about the system dynamics as the vehicle will not respond to these high frequency input variation.

4.2.2 *Hammerhead* trials setup for SI

In light of the discussion in the previous section, experiments were designed that could obtain the best possible data for model development. Ideally, the requirement is to have a completely noise free data which is impossible in a real world environment. The *Hammerhead* is a low speed AUV that swims at about 2 knots. This gives some insight about the sampling period to be chosen. Clearly, too high sampling rate in this case will give no advantage whatsoever. A sampling rate of 1 Hz is thus chosen iteratively which is adequate to obtain ample dynamical information about the system. By the same token, the frequency for the input signal is chosen as 0.1 Hz which was deemed sufficient to excite the interesting modes of the system.

Some common type of excitation signals used in this thesis are uniformly distributed random numbers, pseudo random binary sequence (PRBS), Figure 4.3, and its variants such as multistep, Figure 4.4, and doublet input, Figure 4.5. The multistep inputs are suitable to obtain the step response of the vehicle with various levels of input amplitude. On the other hand, PRBS signal excites the system within a range of frequencies. Additionally, it is easy to repeat to ensure data consistency or for averaging purposes. For depth dynamics, only short duration multistep inputs were used due to the reasons mentioned above. Several experiments had been carried out with a different multistep signal each time, similar to Figure 4.4, applied to the hydroplane. The data obtained from these trials was merged later for model estimation using the SI toolbox in Matlab. The response of the vehicle to these excitation signals will be discussed in the subsequent sections. It should be re-emphasised here that the Matlab environment was used for DAQ during all SI trials. However, since Matlab DAQ abilities are limited, a sequential algorithm was developed that acquire data from various onboard sensors progressively rather than simultaneously. The al-

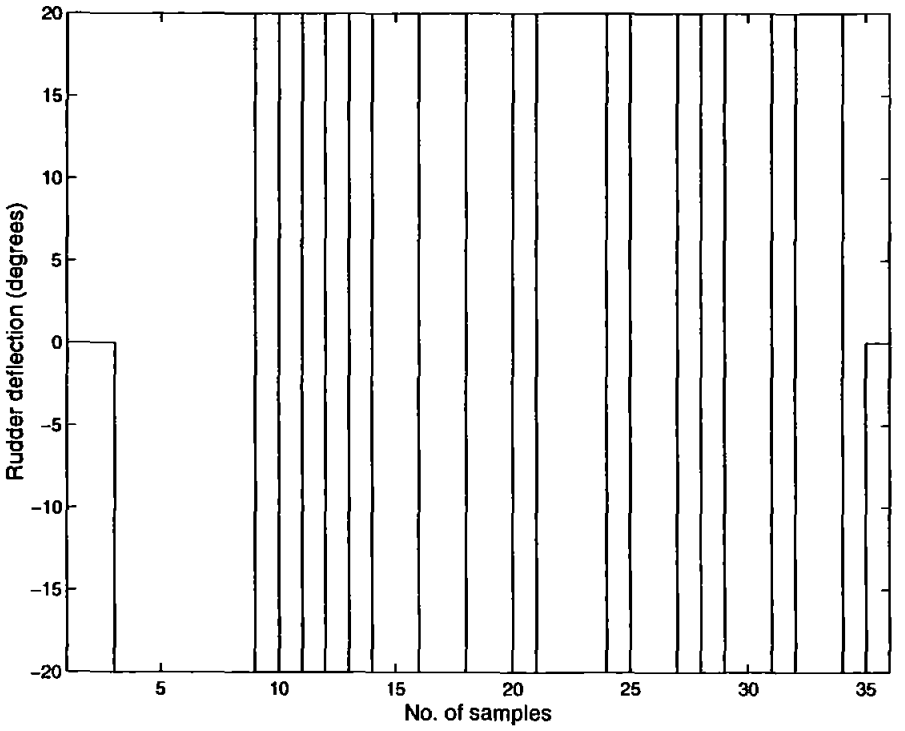


Figure 4.3: 32 length PRBS input for system identification

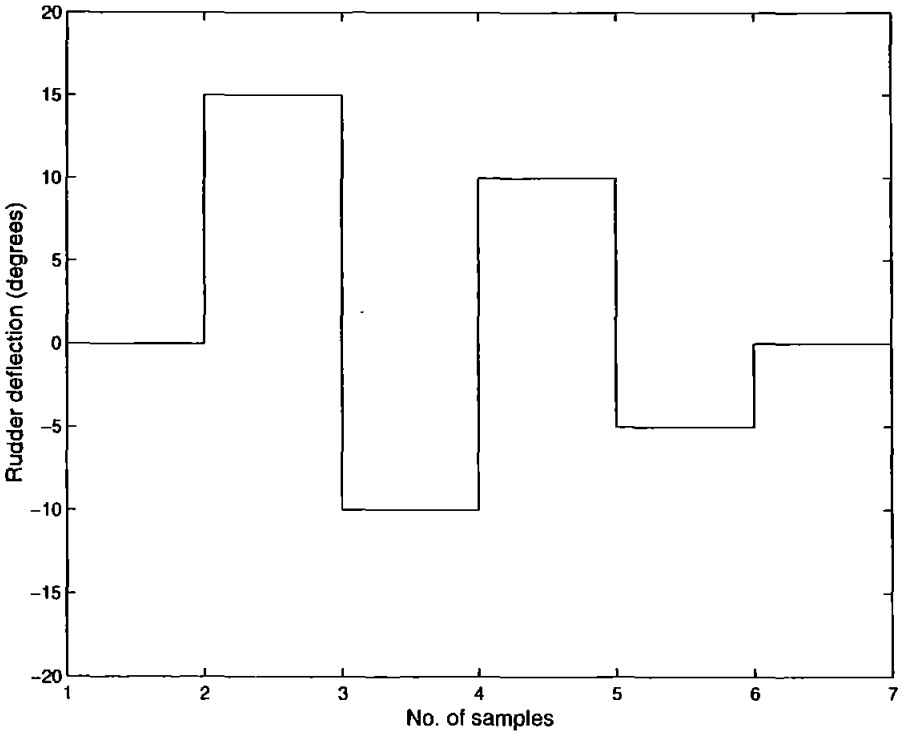


Figure 4.4: Multistep input for system identification

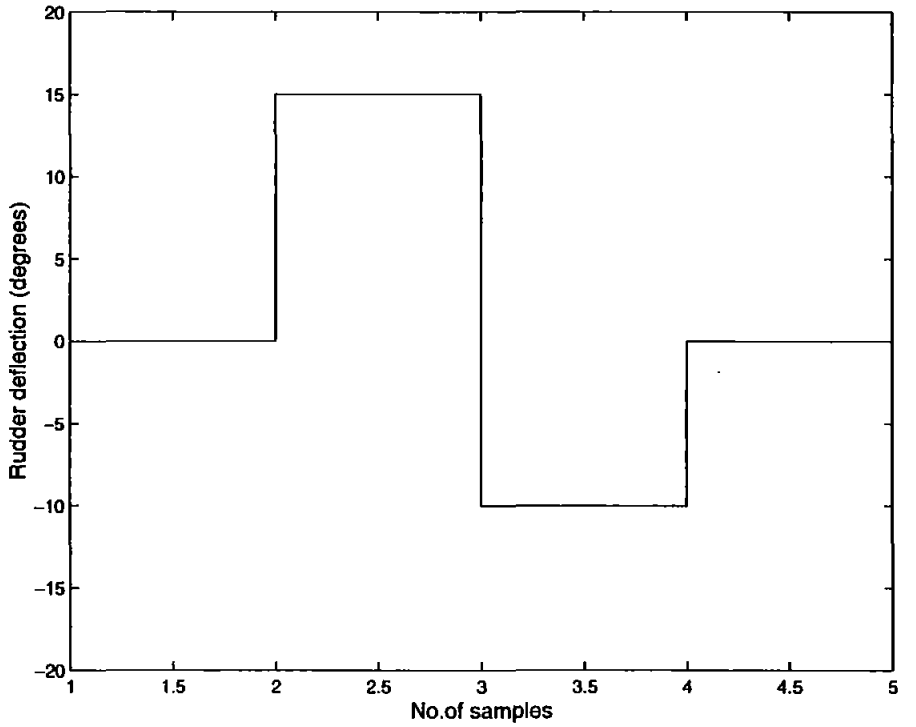


Figure 4.5: Doublet input for system identification

gorithm is listed below showing the order in which identification data was obtained and is explained as follows.

```
Step 1. send input to the control surface
Step 2. while time < specified duration
            read depth sensor
            read TCM2 compass
            read IMU
        end
Step 3. read shaft speed
        read GPS
goto step 1.
```

Each of the excitation signal was applied for a specified duration during which sensors data was collected. For the rudder-yaw channel identification trials, data was acquired from TCM2, IMU, depth sensor, GPS and a shaft speed encoder. Depth channel trials uses the same algorithm except that the GPS data was not monitored. Please note that the GPS and shaft speed encoder have been kept outside of the main loop. This is because the GPS samples at a much slower rate as compared to other sensors and therefore would reduce the overall sampling rate if it was placed inside the loop. The shaft speed encoder was employed here to check the data validity and to make

sure that the vehicle is not slowing down due to low battery power which implies a change in operating condition. The data obtained during this period was therefore successfully separated using the information from the shaft speed encoder and was not used in model identification.

With this configuration, the sampling frequency obtained was $8Hz$. The data was resampled afterwards at $1Hz$ since this frequency was found adequate to control *Hammerhead*. Moreover, it also helps to smooth the data i.e. it acts as a low pass filter. It was observed that during the transmission phase to the onboard actuators, no data could be acquired. This is due to the limitations of Matlab. This problem was circumvented by leaving holes during that interval which represents the missing data. In addition, since there was no feedback from the control surfaces, the transition from one input to the other is approximated as a ramp and appropriate values were inserted. The whole input/output data was later processed and the missing data was interpolated. Figure 4.6(a) shows data set with holes and Figure 4.6(b) depicts the processed data.

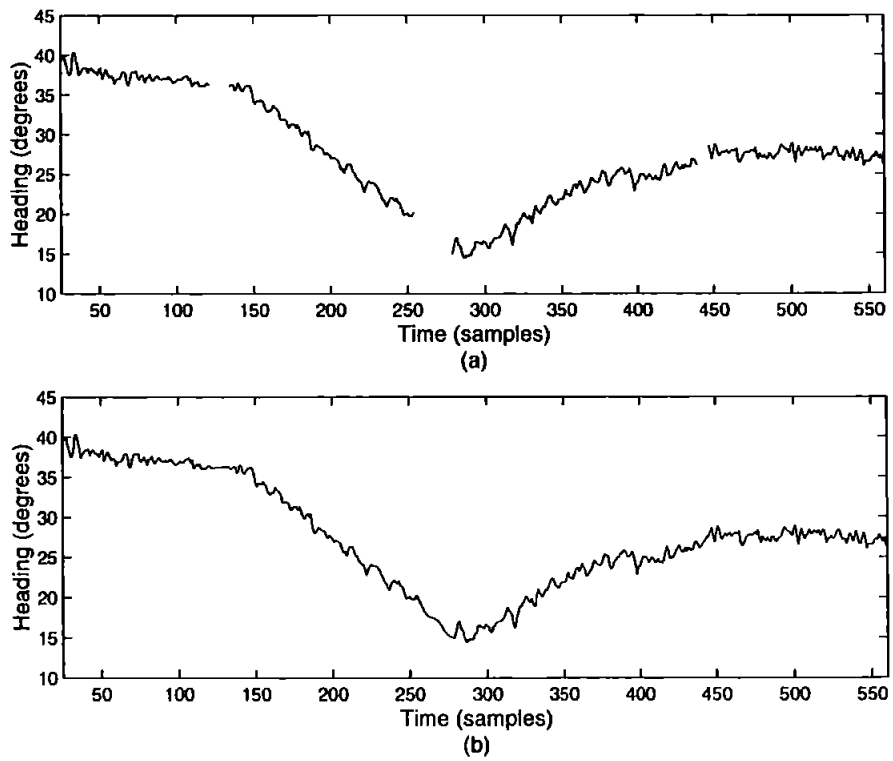


Figure 4.6: Reconstruction of the missing data (a) Original data set and (b) Interpolated data set

4.3 Identification Results

The procedure detailed in Section 4.2.2 was employed to acquire the input/output data from *Hammerhead* necessary for SI. Trials have been performed at South West Water's Roadford Reservoir, Devon, and at Willen Lake in Milton Keynes. Most of the preliminary yaw channel identification trials were conducted at Willen Lake, while additional tests including depth channel analysis was carried out at Roadford Reservoir. The following subsections discuss *Hammerhead* responses to various excitation signals and present SI results when applied to the test trials data (Naeem *et al.*, 2003*b*). As discussed earlier, the yaw channel identification trials were performed just under the surface using a PRBS signal, uniformly distributed random input and various short duration multistep signals whereas only multistep input sequences were used for depth channel which were later merged for SI. Single input single output (SISO) models were developed for both channels. Some cross coupling effects were observed specifically in the depth channel and will be discussed.

4.4 Rudder-Yaw Channel

The input to this channel is the rudder deflections and the output is the vehicle's yaw or heading angle. The heading information is available from the TCM2 and IMU, however, the results presented here are the responses obtained from TCM2 only. In addition, the data sets shown are the original measurements at 8 Hz and have not been filtered or resampled. Three trial results are shown for this channel each with a different input and the responses are the heading, roll and pitch angle. Since the trials for this channel were performed near the surface with the vehicle fully submerged and because there was no cross coupling effects observed in the depth with respect to the change in heading, therefore, the depth data is not shown for this case. A SISO model is developed for this channel shown in the block diagram of Figure 4.7, where $u1$ represents the input rudder deflections, the SISO transfer function is given by G_{11} and the output or heading angle is represented by $y1$.

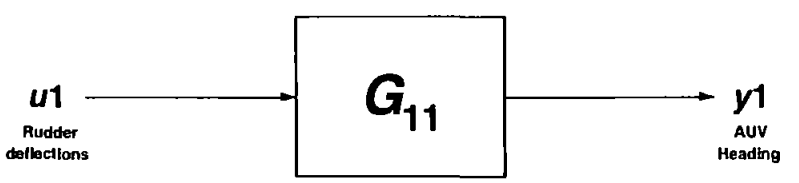


Figure 4.7: Single input single output block diagram of the rudder-yaw channel model

4.4.1 Yaw data analysis

The yaw response of *Hammerhead* to various excitation signals such as PRBS and multistep inputs is presented in this section. Some useful findings from these data sets are also elaborated.

PRBS input

A 32 length PRBS sequence is shown in Figure 4.8(a). The response of *Hammerhead* to this input is also depicted in Figure 4.8(b). Very useful information can be extracted from the heading data. The negative rudder deflection as seen from the figure causes the vehicle to turn clockwise whilst the AUV direction is opposite in case of a positive rudder angle. The turning radius of the *Hammerhead* is an important specification and can be estimated using this data set.

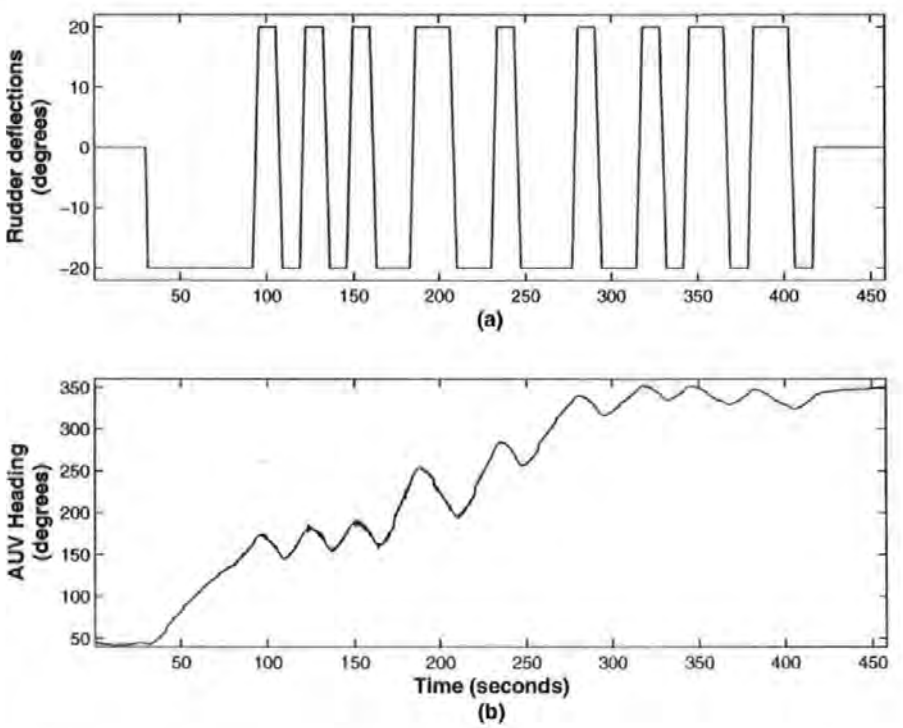


Figure 4.8: Heading response of *Hammerhead* to a PRBS signal applied to the rudder (a) *Hammerhead* heading and (b) Rudder deflections

Let L be the straight line distance travelled by the vehicle in t seconds. If the vehicle velocity is v, then

$$L = vt \tag{4.15}$$

Assume that the vehicle course change in t seconds is α degrees, then the arc length S relating the turning radius of the circle r_a and heading change α, shown in Figure 4.9, is given by

$$S = r_a\alpha \implies r_a = \frac{S}{\alpha} \tag{4.16}$$

where α is in radians.

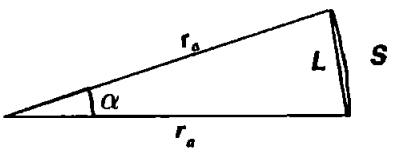


Figure 4.9: Arc length as a function of radius and angle

The arc length S can also be defined in terms of the straight line distance L (Wattenburg, 1996),

$$S = 2r_a \sin^{-1}\left(\frac{L}{2r_a}\right) \tag{4.17}$$

substituting the value of r_a from Equation 4.16 in Equation 4.17 gives

$$r_a\alpha = 2r_a \sin^{-1}\left(\frac{L\alpha}{2S}\right)$$

$$\sin\left(\frac{\alpha}{2}\right) = \frac{L\alpha}{2S}$$

$$\frac{S}{\alpha} = \frac{L}{2\sin\left(\frac{\alpha}{2}\right)}$$

$$\boxed{r_a = \frac{L}{2\sin\left(\frac{\alpha}{2}\right)}} \tag{4.18}$$

Hence the turning radius of the vehicle can be estimated if the straight line distance L and angle spanned by the vehicle in a specified interval are known. In case of *Hammerhead*, the turning radius with an input of 20 degrees to the rudder is found to be approximately 28 metres. However, the full rudder deflection was not utilised in this case, therefore the minimum turning radius should be less than 28 metres. An

earlier study by Dalgleish (2001) on *Hammerhead* predicted the turning radius to be about 22 metres based on observation.

The roll and pitch of the vehicle in Figure 4.10 seems unchanged with respect to the change in heading. The magnitude of the variations in roll and pitch response for this channel is very small and hence could be neglected.

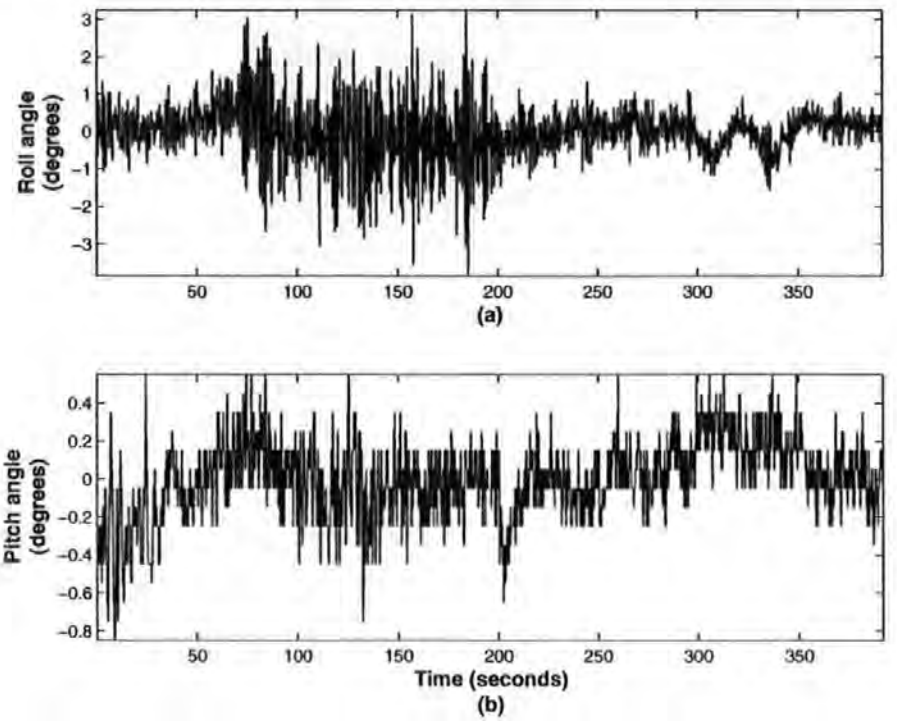


Figure 4.10: Roll and pitch response of *Hammerhead* to a PRBS signal applied to the rudder (a) Roll angle and (b) Pitch angle

Uniformly distributed random input

The excitation signal and the vehicle's response is depicted in Figure 4.11. This provides several step responses of the *Hammerhead* for various levels of input. Looking closely at the response plot, the vehicle course changes for a zero rudder deflection. There could be at least two possible reasons for this type of behaviour. Firstly, the surface currents can push the vehicle without any current compensation or closed loop control. Secondly, since the proximity sensors were not installed at the time of the experiments, the rudder was being initialised by observation and hence the exact position of the rudder was uncertain. The roll and pitch in Figure 4.12 remains unaltered as before with respect to the heading and therefore were not considered in

the modelling process.

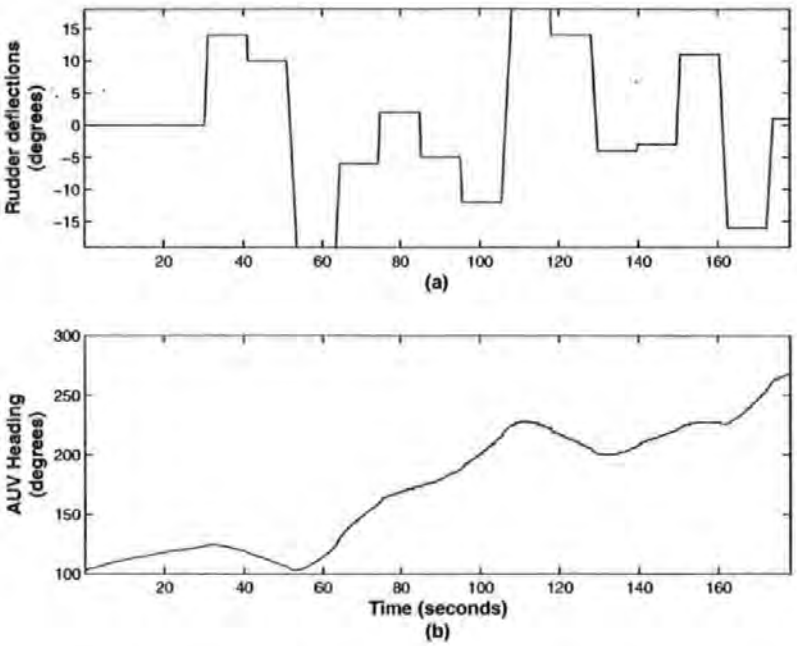


Figure 4.11: Heading response of *Hammerhead* to a uniformly distributed random signal applied to the rudder (a) *Hammerhead* heading and (b) Rudder deflections

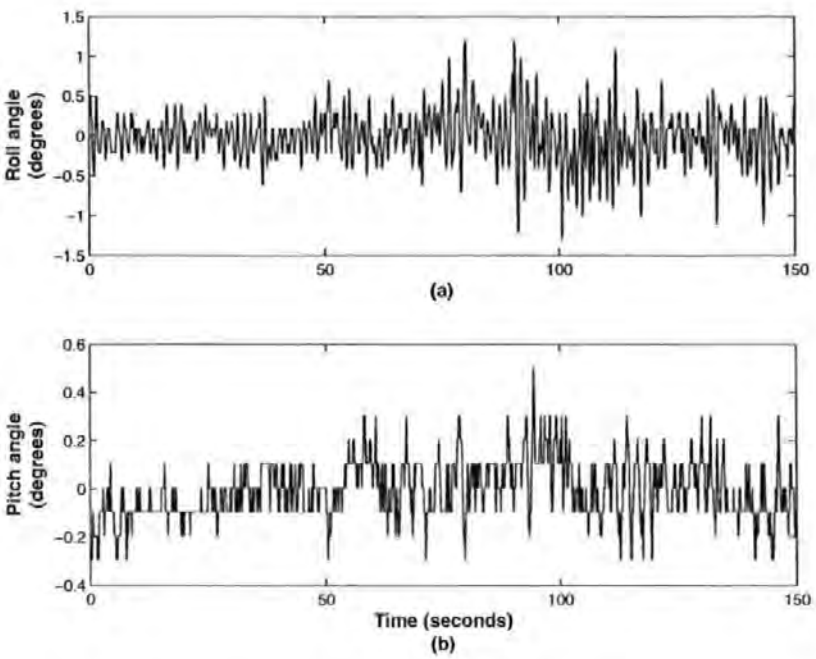


Figure 4.12: Roll and pitch response of *Hammerhead* to a uniformly distributed random signal applied to the rudder (a) Roll angle and (b) Pitch angle

Multistep input

Finally, the multistep input in Figure 4.13(a) was used to excite the *Hammerhead* dynamics and the response was recorded in Figure 4.13(b). Again the vehicle heading changes for zero rudder deflection for the first 15 seconds followed by a change of heading angle in response to two successive positive rudder inputs of magnitude 5 and 15 degrees. A negative input was then applied to the rudder causing the vehicle to turn closkwise. Similar observations could be made in the last 15 seconds when the rudder angle is zero, however, the vehicle is not moving in a straight line as discussed earlier.

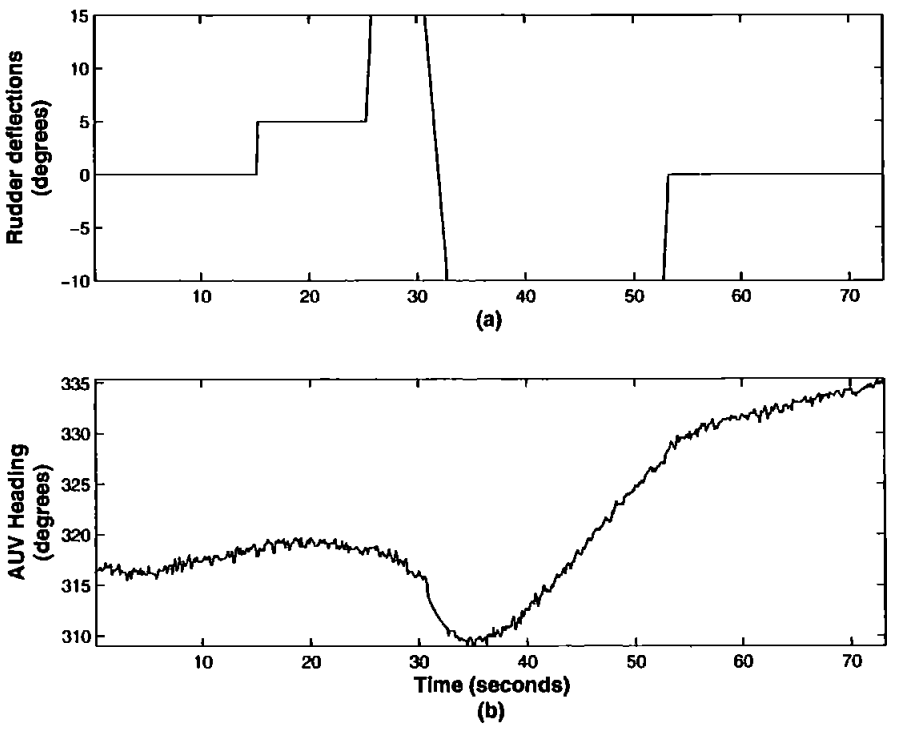


Figure 4.13: Heading response of *Hammerhead* to a multistep input applied to the rudder (a) *Hammerhead* heading and (b) Rudder deflections

Figure 4.14 depicts the roll and pitch movements of the vehicle for the multistep input experiment and observed to be unaffected.

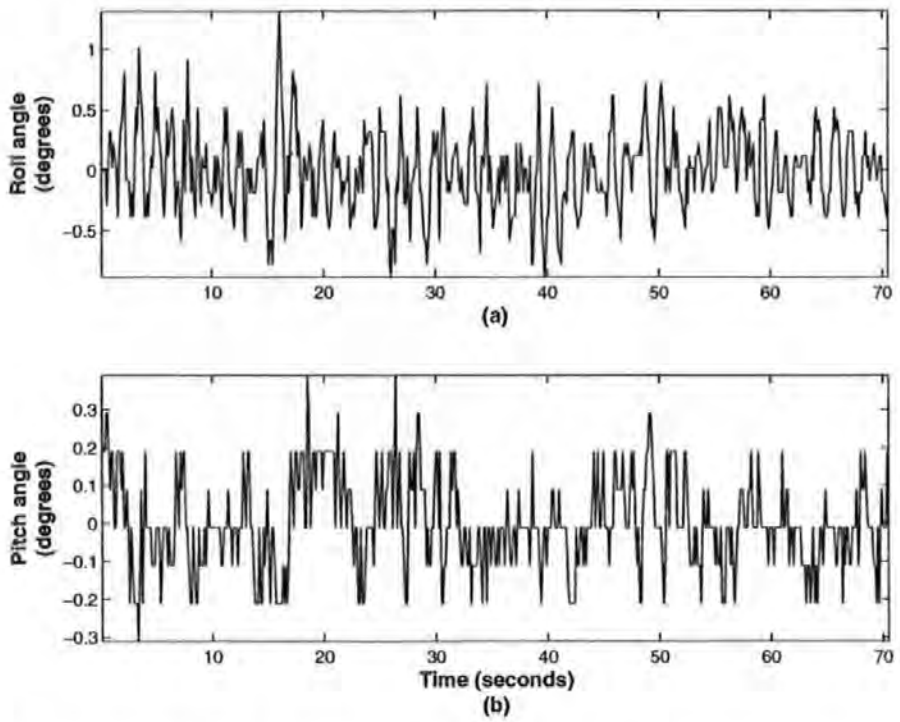


Figure 4.14: Roll and pitch response of *Hammerhead* to a multistep input applied to the rudder (a) Roll angle and (b) Pitch angle

4.4.2 Modelling of rudder-yaw channel

Once suitable data sets were selected, attention was turned towards estimating a model that could best replicate the systems dynamic behaviour. All available measurements were pre-filtered and resampled at 1 Hz before any model parameters could be identified. Due to this, most of the high frequency contamination was eliminated and hence it was decided to extract an ARX or state space model without modelling the noise separately. For model order selection, an approach called Akaike's final prediction error (FPE) has been adopted (Ljung, 1999). According to Akaikes theory, in a collection of different models, choose the one with the smallest FPE. A plot of the FPE versus the model order is depicted in Figure 4.15 for the yaw channel. Clearly, the FPE is minimum for a third order model, however, the validation tests suggest a minor increase in data fit as compared to a second order model. An ARX(221) model has thus been extracted and is given by Equation 4.19

$$G(q) = \frac{-0.04226q^{-1} + 0.003435q^{-2}}{1 - 1.765q^{-1} + 0.765q^{-2}} \tag{4.19}$$

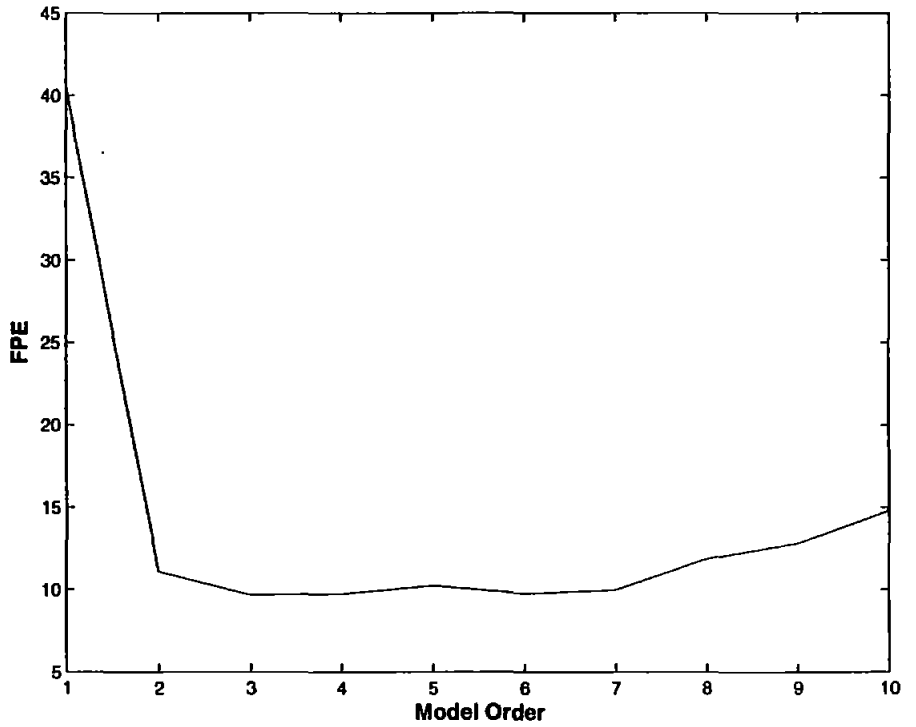


Figure 4.15: FPE versus model order for the selection of the rudder-yaw channel model

where q^{-1} is the delay operator. The state space equivalent of this model is

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \tag{4.20}$$

$$y = \mathbf{C}\mathbf{x} + \mathbf{D}u \tag{4.21}$$

where $\mathbf{x}$ is a state vector, u represents the input and y is the output which in this case are the rudder deflections and vehicle heading respectively. The matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ and $\mathbf{D}$ are given by

$$\mathbf{A} = \begin{bmatrix} 1.765 & -0.765 \\ 1 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} -0.04226 & 0.003435 \end{bmatrix} \text{ and}$$

$$\mathbf{D} = 0$$

4.4.3 Model validation

Validation of the estimated model is the final step in an SI process. Various techniques are employed to measure the model quality and its capability to predict accurately the measured response. Correlation tests are performed to validate if all the interesting vehicle dynamics have been captured by the model. On the other hand, cross validation test is performed to gauge the predicting capacity of the model. In this test, data not used for SI is applied to the model and the simulated output is compared with the observed response. The correlation tests of the rudder-yaw channel model were performed and the results are shown in Figure 4.16.

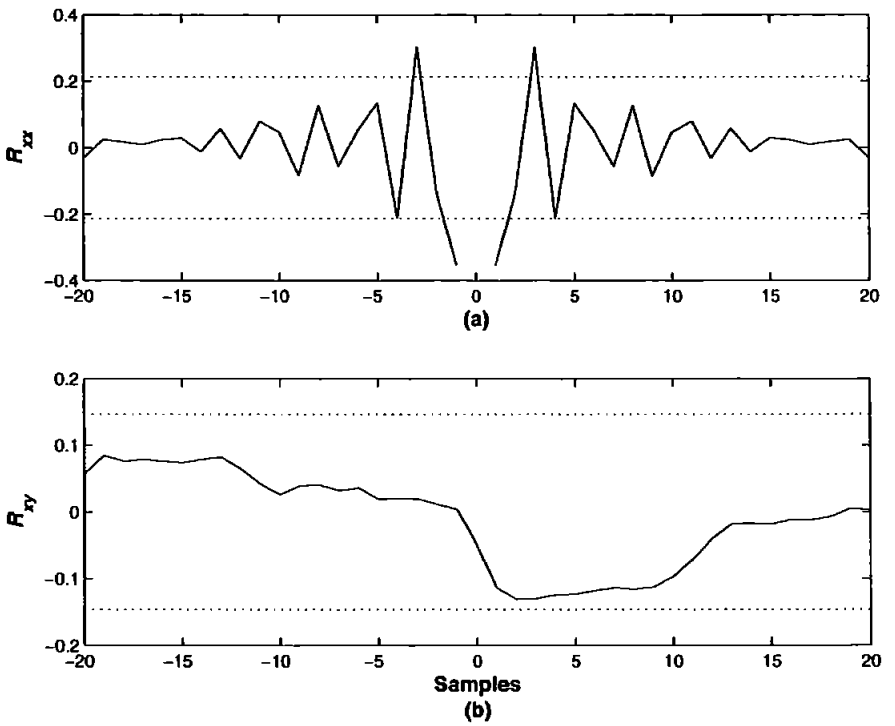


Figure 4.16: Correlation tests for rudder-yaw channel model (a) Autocorrelation of residuals and (b) Cross correlation of residuals and the input

Clearly, the auto and cross correlation functions are within the confidence bands indicating that the extracted model fits well with the measured data and thus deemed adequate for further analysis. Two cross validation tests are next performed for this channel and are shown in Figures 4.17 and 4.18. The simulated outputs as seen from the figures match reasonably well with the measured outputs. There are some discrepancies though evident from the figures due to the effect of surface currents on different data sets during the trials. A higher order model will not give any significant improvement over the estimated model and therefore robust controllers need to be

developed for this model which should be able to cope with any discrepancies and disturbances.

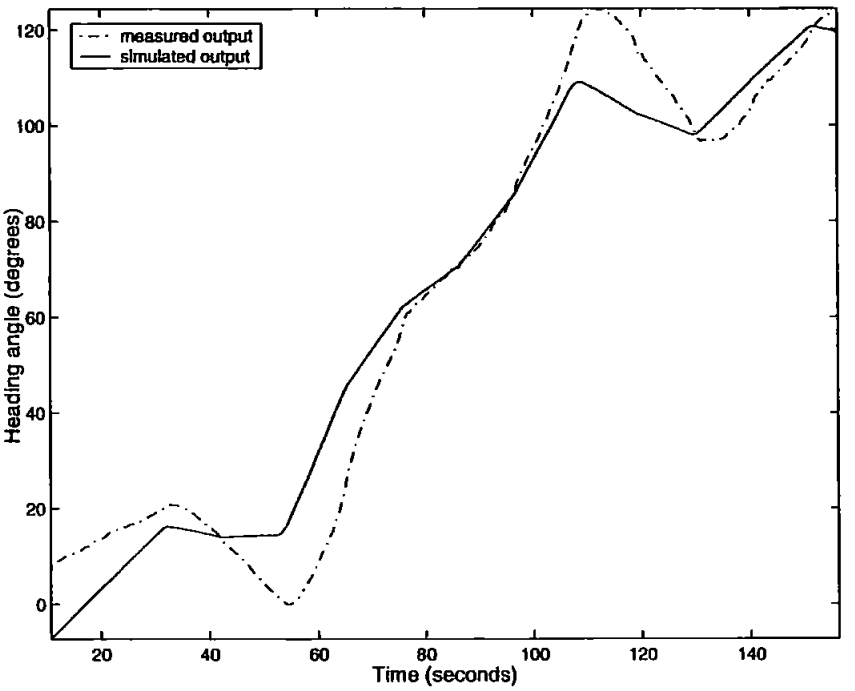


Figure 4.17: Cross validation test for rudder-yaw channel

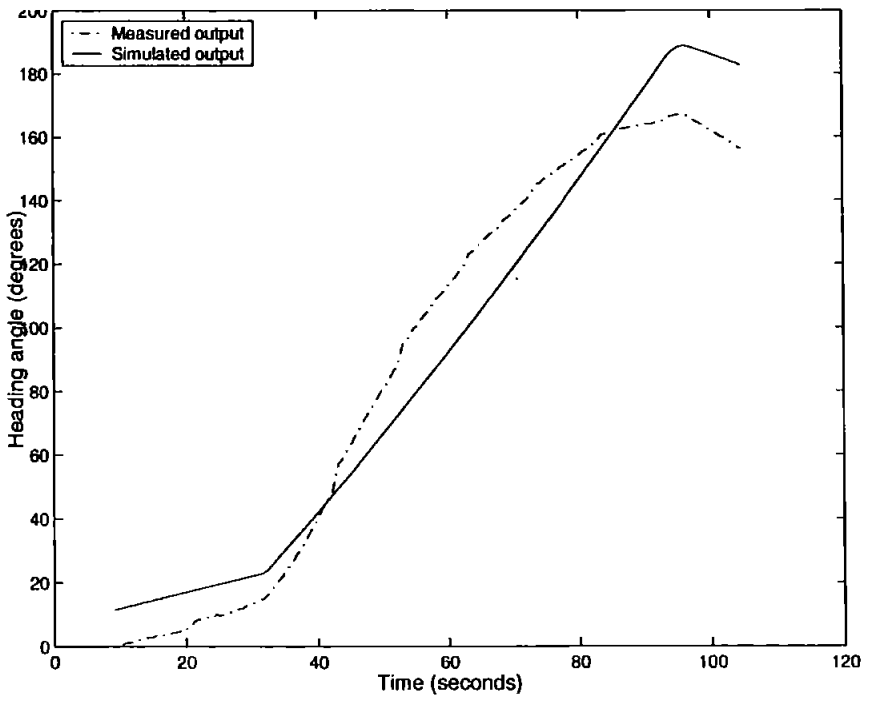


Figure 4.18: Another cross validation test for rudder-yaw channel

4.4.4 Model analysis

The SI approach is a black box modelling technique meaning that no physical quantities are directly involved in this process in contrast to mathematical modelling. All that is of interest is the cause and effect phenomena and then identifying the black box in between (see Figure 4.7), that can reproduce the measured system output as closely as possible for the same input. Some insight can be gained into systems behaviour by analysing the estimated model. The coefficients of the model, though, do not have any direct physical interpretation but they are vital in studying the nature of the system. The numerator coefficients, for instance, provide the zeros of the plant. For many applications, the plant needs to be minimum phase i.e. all zeros must lie within the unit circle. The denominator coefficients, on the other hand, determine the pole locations in the z-plane. A pole outside the unit circle indicates an unstable system, therefore the system needs to be stabilised through closed loop control. Clearly, a non-minimum phase plant, when inversed, will swap the location of poles and zeros, in which case, the non-minimum phase zeros will become the undesirable unstable poles.

The pole zero plot of the rudder-yaw channel is shown in Figure 4.19, which clearly shows that this is a minimum phase system. However, the plant is type 1 or marginally stable due to the presence of a pole on the unit circle.

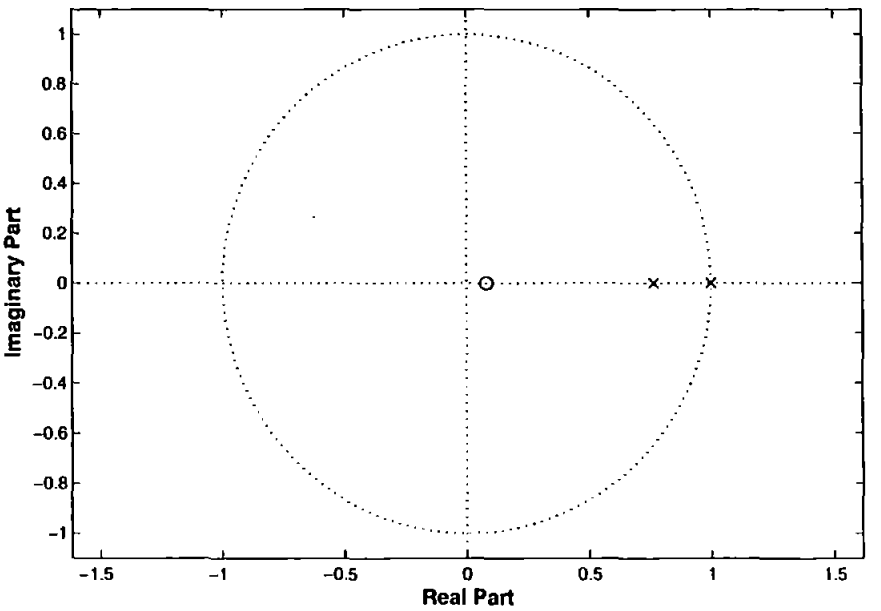


Figure 4.19: Pole zero plot for the rudder-yaw channel model

The step response in Figure 4.20 also reveals the occurrence of that pole which provides an integrator type output to the overall system. The stair like effect in the step response plot indicates a discrete time plant with a sampling period of 1 second.

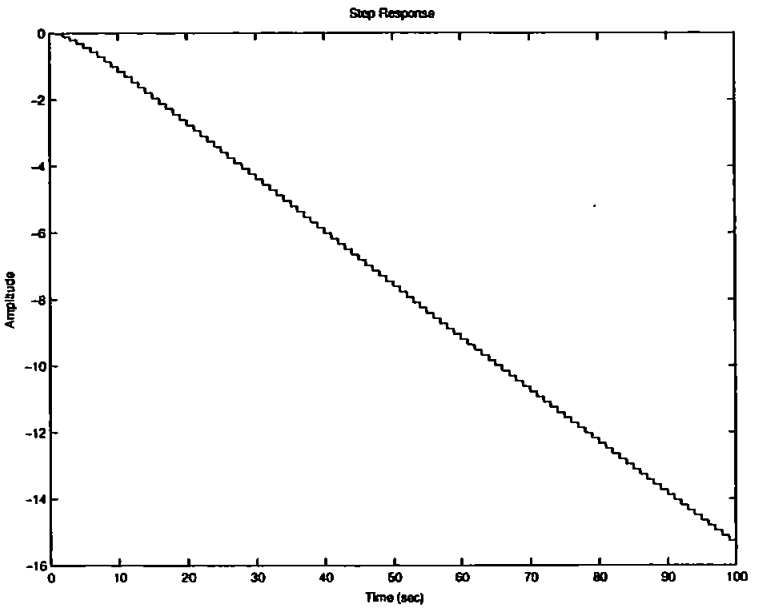


Figure 4.20: Step response of the rudder-yaw channel model

4.5 Hydroplane-Depth Channel

In this section, the *Hammerhead* hydroplane-depth channel is modelled. Data sets were presented and scrutinized that were acquired during the trials at Roadford Reservoir. It has been found that experimenting with the *Hammerhead* depth dynamics was much more involved as compared to the rudder-yaw channel. One of the major difficulties is that the umbilical, which was used as a means of communication with the vehicle, was observed to be affecting its motion considerably. The cable itself was negatively buoyant and a significant amount of length in the water meant too much weight causing the vehicle to sink. Moreover, during the preliminary depth channel identification trials, a leak was found apparently at the point where the cable was attached through a gland connector near the tail of *Hammerhead*. Fortunately, a leak detector was installed on the *Hammerhead* which precludes any further damage to the vehicle and was safely recovered. The worst scenario occurred when the cable was completely severed by the propeller resulting in the trials being aborted. Luckily, the vehicle was recovered again, thanks to the fresh water reservoir that prevents any conducive action on the exposed contacts of the switch of the main thruster. A new

cable was then procured, installed and depth trials were performed at a later date.

The input to this channel is the hydroplane deflections while the output is the depth of the vehicle taken from a pressure transducer. The transfer function block diagram is shown in Figure 4.21, where G_{22} is a transfer function that maps the input (control surface angle) to the output space (depth).

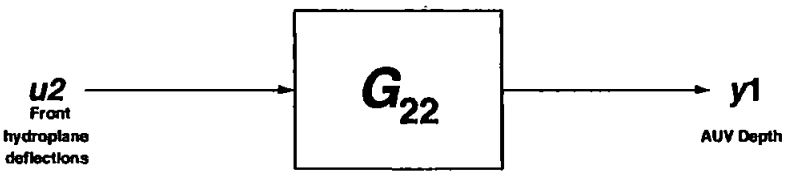


Figure 4.21: SISO block diagram of the *Hammerhead* hydroplane-depth channel

Only multistep inputs were employed to excite the depth dynamics of the vehicle due to the reasons mentioned before. The GPS data was not monitored since the GPS signal gets attenuated under water due to scattering and thus cannot provide any useful information. In addition, since the vehicle goes out of sight when in deep water, it is difficult to follow on a pursuit boat. For this reason, the depth and heading data was being continuously monitored. A hand held GPS on the surface provided the heading angle and was compared with the vehicles heading. This information was proved to be viable for following the vehicle during submerged operations.

Figure 4.22 depicts the depth trials of *Hammerhead* vehicle at Roadford Reservoir. The umbilical cable and the pursuit boat can also be seen in the figure. The vehicle was allowed to swim freely in 6 DOF, however, only a single input (hydroplane) was manipulated and the heading, depth, roll and pitch data were analysed and recorded.

Figure 4.22: *Hammerhead* depth channel identification trials at Roadford Reservoir, showing the pursuit boat and the umbilical

4.5.1 Depth data analysis

For each multistep input, the experiment was repeated at least three times and data acquired was averaged and resampled at 1Hz. Inconsistent data sets were rejected and were not used in model estimation. Figure 4.23(a) shows one of the several multistep inputs and the depth response is presented in Figure 4.23(b). For zero control surface deflection, the vehicle swims on the surface and dives with a positive deflection angle applied to the hydroplane. The vehicle goes down to 1 metre of depth before the control surface position was switched to the negative side causing the vehicle to resurface. A positive excitation input again with an increased duration than the first pushed the vehicle down to approximately 6 metres. Finally, a negative input of a smaller magnitude and longer duration helped *Hammerhead* to recover on the surface. It should be noted in the response plots that there is a small delay between the input and output response times. This is because the pressure sensor is installed on the back side near the tail of the vehicle. With the same magnitude of positive and negative inputs to the hydroplane, it is evident from the response plot that the *Hammerhead* diving rate is much faster than the surfacing rate. For instance, the diving rate with an input magnitude of +15 degrees is approximately 0.6 metres per second, while the vehicle surfaces with the same level of input signal

at a rate of about 0.2 metres per second. This behaviour enlightens the slow dynamic characteristics of *Hammerhead* and signifies the effect of the umbilical on the vehicle dynamics. In this case, the vehicle had to carry the weight of the negatively buoyant cable thus influencing the depth data quality.

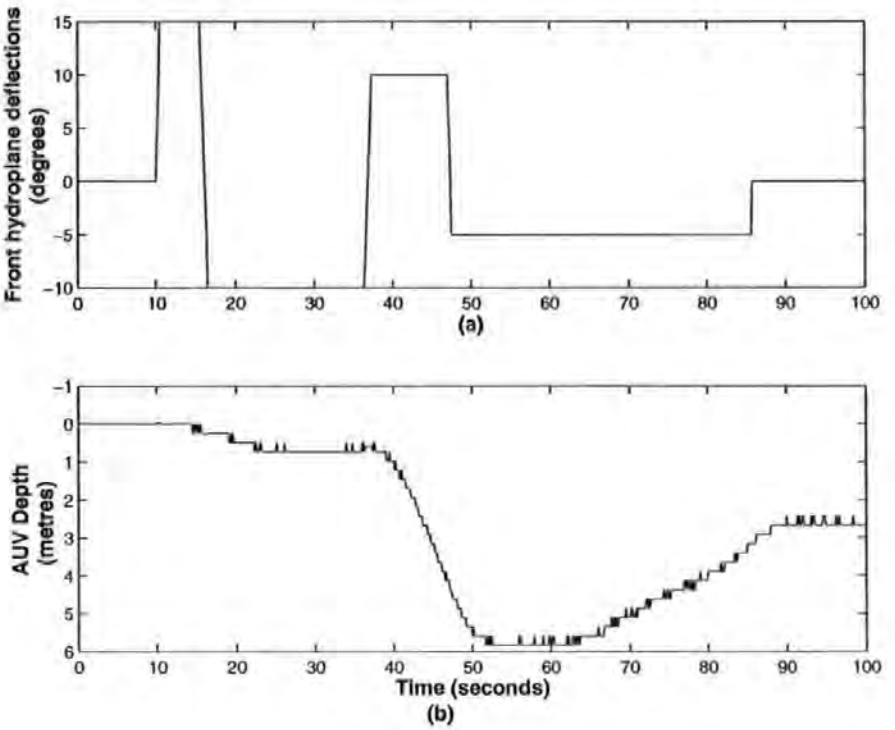


Figure 4.23: Depth response of *Hammerhead* to a mulistep input (a) *Hammerhead* depth and (b) Input hydroplane deflections

The heading plot in Figure 4.24(a) reveals interesting dynamics of the vehicle as the heading angle changes abruptly during the initial diving phase. However, it is much more stable when the vehicle is fully submerged. The cause of this phenomenon was unknown at that point and the umbilical was thought to be the source of this erratic behaviour. The pitch angle is also shown in Figure 4.24(b) indicating decreasing pitch angles for diving and opposite for surfacing. The maximum pitching angle of *Hammerhead* in this response plot is approximately -20 degrees. The roll in Figure 4.24(c) varies between 0 and -1 degrees during the whole mission duration. The magnitude of roll variations is deemed negligible and thus will not be considered any further.

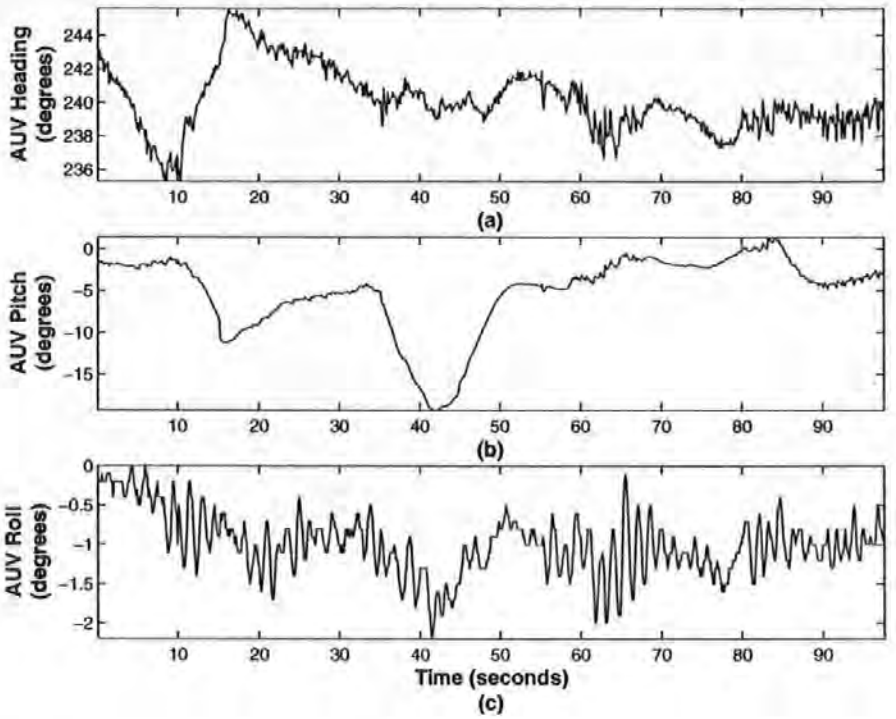


Figure 4.24: Euler angle responses of *Hammerhead* to a multistep input applied to the hydroplanes (a) *Hammerhead* heading (b) Pitch angle and (c) Roll angle

Another multistep input and the *Hammehead* depth response is shown in Figure 4.25(a) and 4.25(b) respectively. The response is similar to the previous one except that there are now three positive step signals applied to the control surface and the vehicle made diving manoeuvres in response to the inputs while submerged at a depth of about 10 metres. The vehicle resurfaces finally at the end of the run. The heading plot shown in Figure 4.26(a) bears similar abrupt characteristics at approximately 15 seconds when diving. The vehicles pitch angle in Figure 4.26(b) saturates at -20 degrees since this is the maximum output from the TCM2. The maximum positive pitch in this case is about 10 degrees. The roll data in Figure 4.26(c) shows some change during the diving manoeuvre. However, it is nearly uniform when the vehicle is fully submerged.

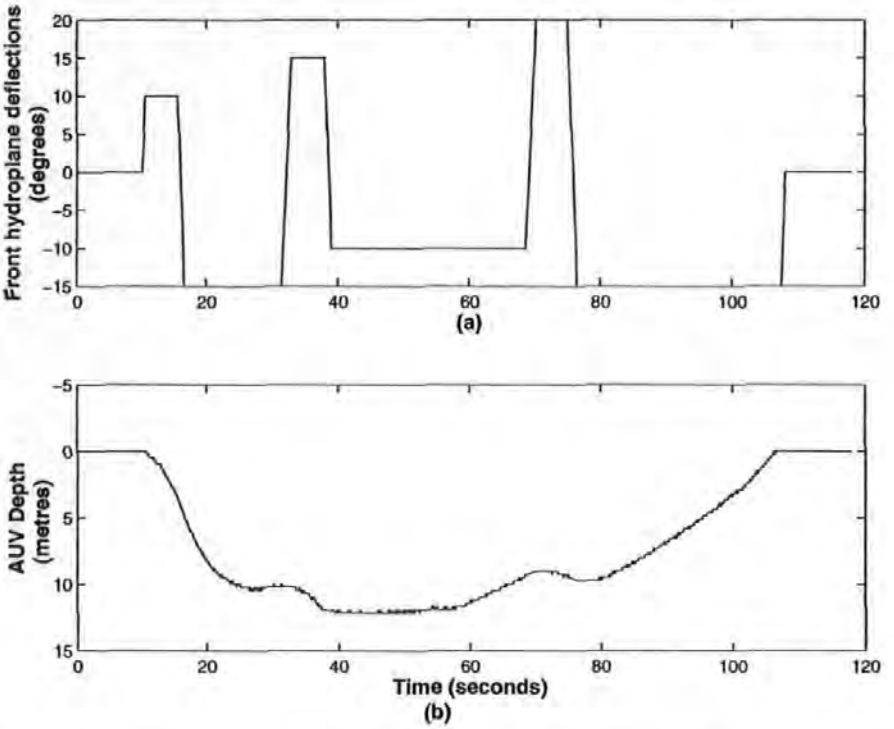


Figure 4.25: Depth response of *Hammerhead* to a mulistep input (a) *Hammerhead* depth and (b) Input hydroplane deflections

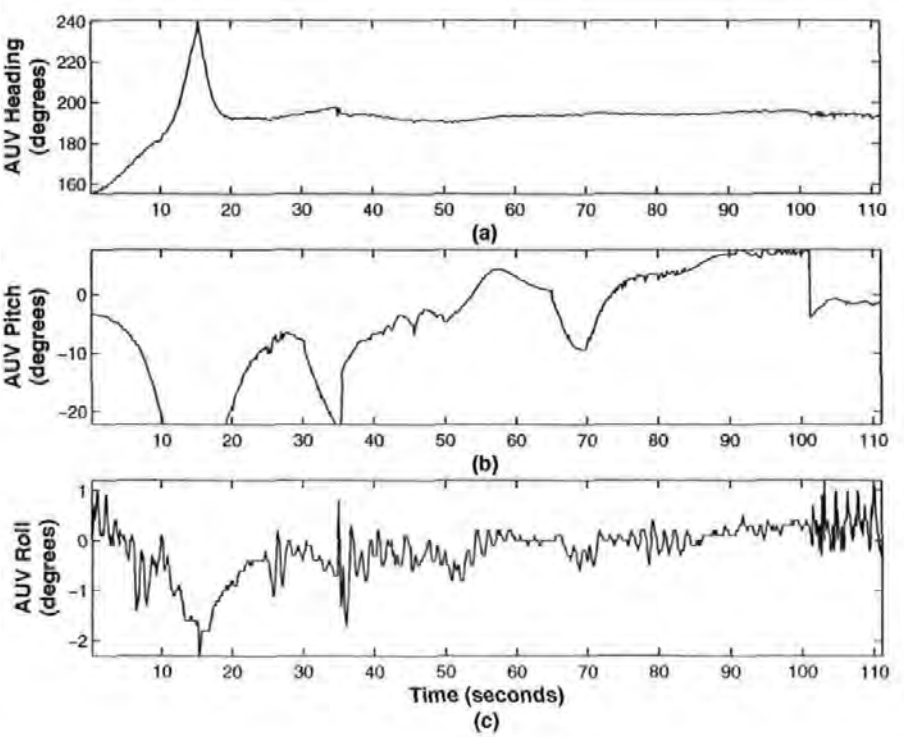


Figure 4.26: Euler angle responses of *Hammerhead* to a multistep input applied to the hydroplanes (a) *Hammerhead* heading (b) Pitch angle and (c) Roll angle

4.5.2 Modelling of hydroplane-depth channel

The *Hammerhead* data obtained from the depth trials reveal some cross coupling effects between the depth and heading angle. A multivariable model therefore should be the ideal choice. However, exploiting the fact that the heading angle does not vary significantly when the vehicle is fully submerged, a SISO model involving only hydroplane deflections and depth has been developed. Several data sets containing these parameters were collected and suitable data were averaged, resampled and then merged to estimate the model coefficients. The performance index J in this case for the merged data sets is taken as the sum of square of the difference between the model output and individual data sets. This is given by Ljung (1999) as

$$J = \sum_{D^1}(y_1(t) - \hat{y}(t|\Gamma))^2 + \sum_{D^2}(y_2(t) - \hat{y}(t|\Gamma))^2 + \cdots + \sum_{D^n}(y_n(t) - \hat{y}(t|\Gamma))^2 \quad (4.22)$$

where y_n represents the measurements from the nth data set, Γ is a vector containing model coefficients and D^n is the number of points in the nth data set. For model order selection, a plot of FPE versus model order for the depth channel is shown in Figure 4.27.

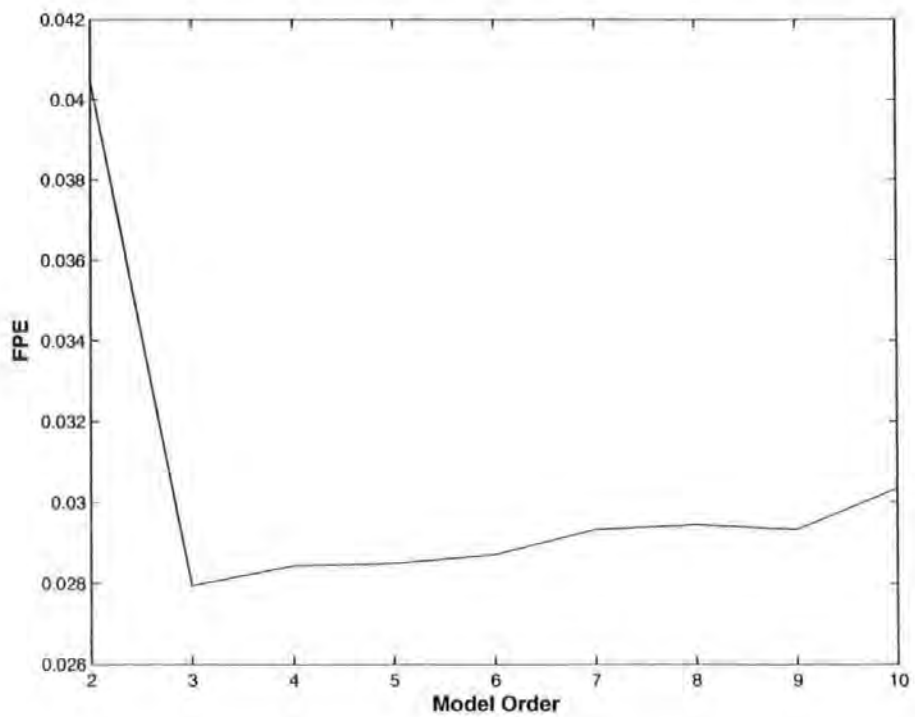


Figure 4.27: FPE versus model order for the selection of the hydroplane-depth channel model

A third order model seems to be the ideal choice since the FPE is minimum at this value. However, a fourth order ARX model was chosen iteratively which gives the best fit between measured and model predicted outputs. Equation 4.23 below presents the ARX(441) model of the depth dynamics of *Hammerhead*

$$G(q) = \frac{0.002681q^{-1} - 0.00327q^{-2} - 0.0007087q^{-3} + 0.001322q^{-4}}{1 - 3.6773q^{-1} + 5.0839q^{-2} - 3.1348q^{-3} + 0.72826q^{-4}} \tag{4.23}$$

The state space equivalent of the above model is given by Equations 4.20 and 4.21 where the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ and $\mathbf{D}$ are given by

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.72826 & 3.1348 & -5.0839 & 3.6773 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0.0026807 & 0.0065881 & 0.009889 & 0.012597 \end{bmatrix}^T$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \text{ and}$$

$$\mathbf{D} = 0$$

where u is the hydroplane deflections and y represents the depth of the vehicle.

4.5.3 Model validation

The model was verified by analysing the correlation tests and time domain cross validation. The ACF and CCF are depicted in Figure 4.28(a) and 4.28(b) respectively. Both functions are well within the confidence intervals and hence verify the criteria in Equations 4.13 and 4.14. Two data sets were then chosen for cross validation test which were not used for parameter estimation. The results of these tests are shown in Figures 4.29 and 4.30. Clearly, the measured and simulated outputs are in harmony i.e. the model output is closely following the measured response. This model of *Hammerhead* will be used to develop depth controllers in simulations and real time in the forthcoming chapters.

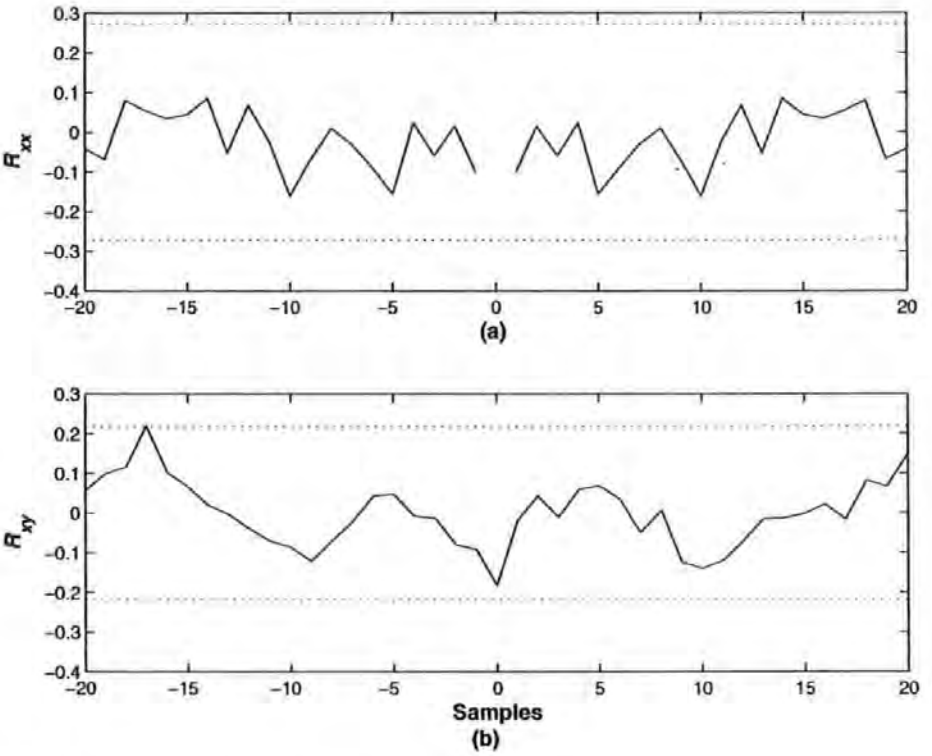


Figure 4.28: Correlation tests for hydroplane-depth channel model (a) Autocorrelation of residuals and (b) Cross correlation of residuals and the input

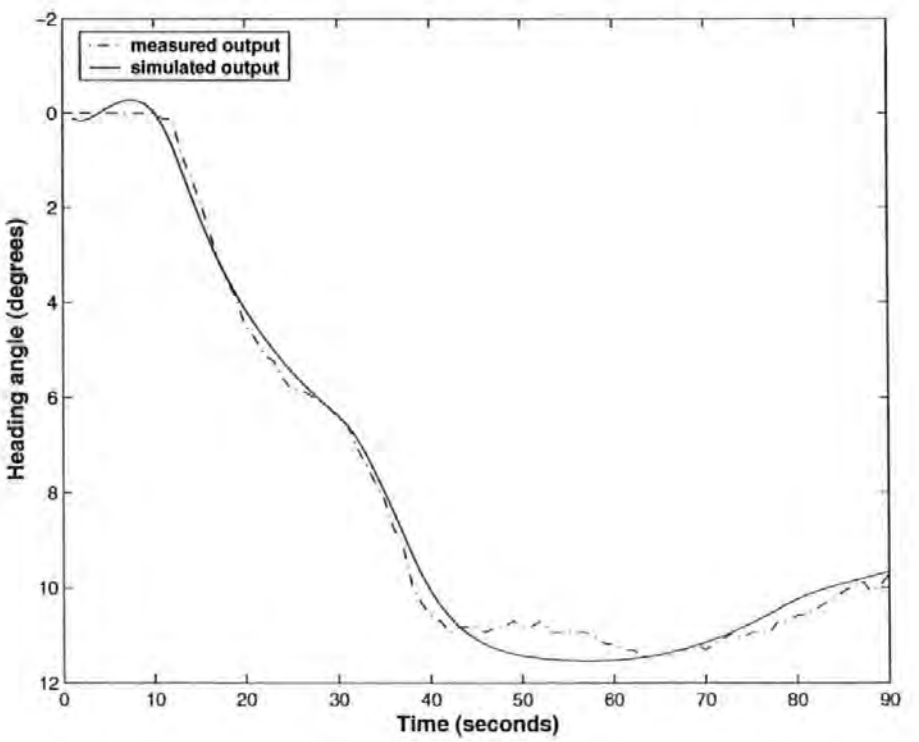


Figure 4.29: Cross validation test for hydroplane-depth channel model

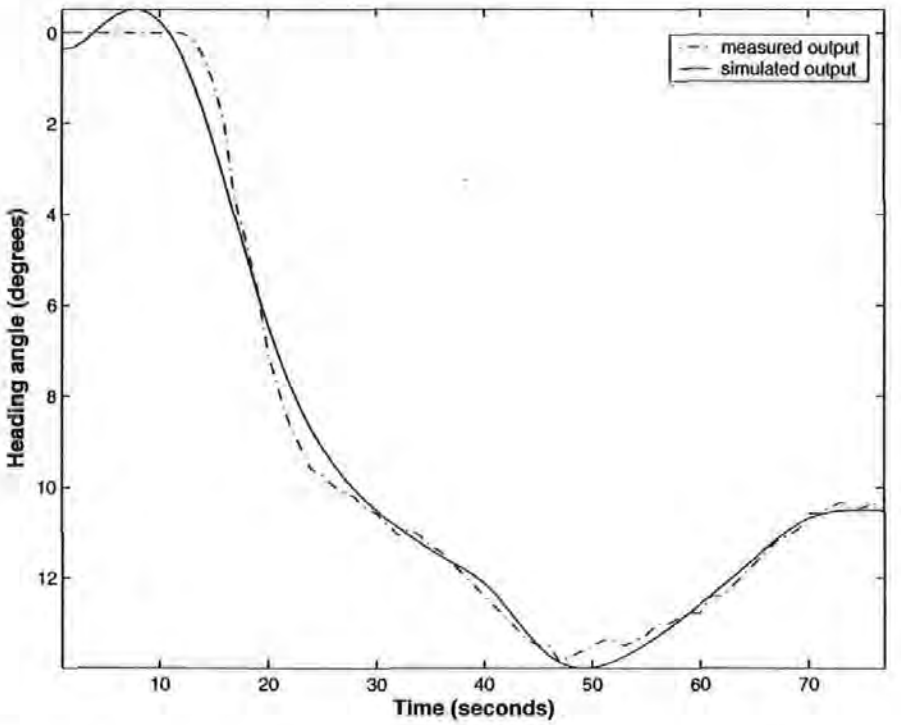


Figure 4.30: Another cross validation test for hydroplane-depth channel model

4.5.4 Model analysis

The model is analysed finally to gain some depth into systems behaviour without having to consider the physical parameters of the vehicle. Similar to rudder-yaw channel, the model is studied for any particular pole zero locations that might reveal some of the system characteristics. Factoring the numerator and denominator of the transfer function model in Equation 4.23 gives

$$G(q) = \frac{0.0064295q^{-1}(1-0.9457q^{-1})(1+0.5561q^{-1})(1-0.09531)}{(1+0.7192q^{-1})(1-0.8048q^{-1})(1-0.9517q^{-1})(1-0.9975q^{-1})} \tag{4.24}$$

which clearly shows the location of poles and zeros in the z-plane. Figure 4.31 depicts the pole zero plot with circles representing the zero locations and poles locations are marked by a cross. Since all the zeros are residing within the unit circle, the system is minimum phase. There is a pole location near to boundary $|z| = 1$ indicating that this is very close to a marginally stable system with very slow dynamics and a step response that resembles that of an integrator type for small step durations as shown in Figure 4.32. Additionally, no oscillatory (complex conjugate) poles are contained

in the model, however, there is an almost overlap of a pole and zero locations apparent in Figure 4.31 which if ignored would not give satisfactory performance.

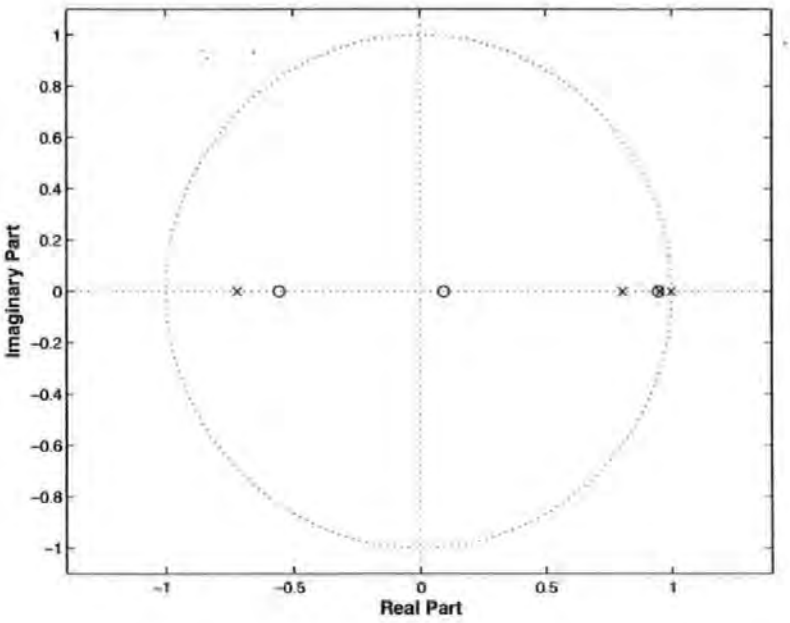


Figure 4.31: Pole zero plot for the hydroplane-depth channel model

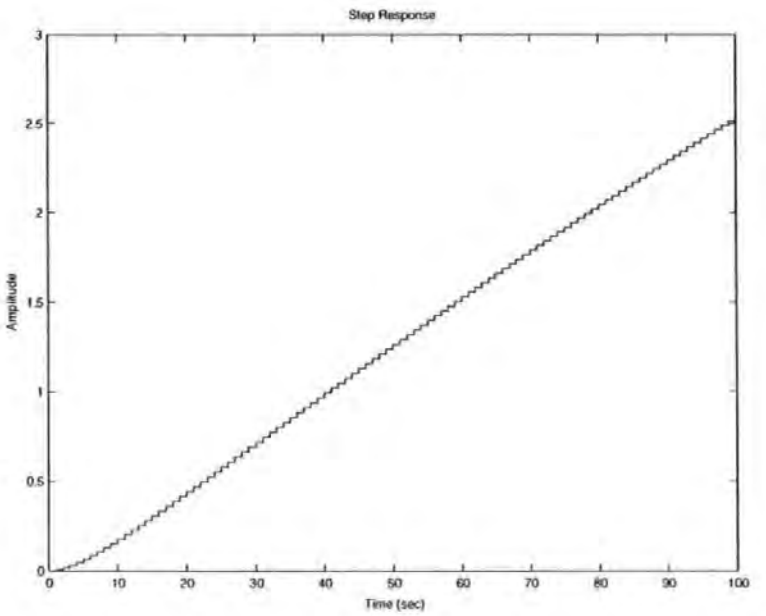


Figure 4.32: Step response of the hydroplane-depth channel model

4.6 Disturbance Modelling

So far the plant considered is completely noise free i.e. no disturbance or noise is taken into account or has been modelled. The high frequency noise present in the data was filtered out before model estimation. However, a controller performance can truly be gauged in presence of external disturbances. In case of underwater vehicles, environmental forces acting on the vehicle tends to vary vehicle's orientation, position and velocity. These include wave disturbances, surface currents and winds with low and high frequency content. The nature of these unwanted forces acting on the vehicle could be both additive and multiplicative. However, for most marine control applications, the disturbances effect can be approximated as additive to the dynamics (Fossen, 1994). In general, there are two types of contaminations considered in an underwater vehicle autopilot design.

High frequency noise which is produced by the sensors and affecting almost every data sample. The data acquired for SI shown in this chapter has this phenomenon quite evident. It is clear that when this noisy data is used as a feedback to the controller, it can cause high frequency control surface movements. To avoid this, the data is treated by a low pass filter before it is passed to the controller.

Low frequency disturbances such as wind, waves generated by wind and ocean current. These cannot be rejected as before therefore the controller design should include the effects of these disturbances. The modelling is therefore imperative to simulate their effect on the overall performance. A sea current model has been devised by Marshfield (1992) and is calculated along the translational axes as follows

$$\begin{aligned}
X_{current} &= X_{velocity}\left[\cos(\psi)\cos(\theta)\right] + Y_{velocity}\left[-\sin(\psi)\cos(\theta)\right] \\
&+ Z_{velocity}\left[\sin(\theta)\right] \\
Y_{current} &= X_{velocity}\left[-\sin(\psi)\cos(\phi) + \cos(\psi)\sin(\theta)\sin(\phi)\right] \\
&+ Y_{velocity}\left[-\cos(\psi)\cos(\phi) - \sin(\psi)\sin(\theta)\sin(\phi)\right] \\
&+ Z_{velocity}\left[-\cos(\theta)\cos(\phi)\right] \\
Z_{current} &= X_{velocity}\left[\sin(\psi)\sin(\phi) + \cos(\psi)\sin(\theta)\cos(\phi)\right] \\
&+ Y_{velocity}\left[\cos(\psi)\sin(\phi) - \sin(\psi)\sin(\theta)\cos(\phi)\right] \\
&+ Z_{velocity}\left[-\cos(\theta)\cos(\phi)\right]
\end{aligned} \tag{4.25}$$

where $X_{current}, Y_{current}$ and $Z_{current}$ are the components of current in x, y and z directions respectively and ϕ, θ and ψ represents the Euler angles or orientation

of the vehicle in earth-fixed frame.

Herein the sea currents are considered only and a two-dimensional model described in terms of the average current velocity $V_{current}$ and direction of current β is given by Fossen (1994) as

$$\dot{x}_{current} = V_{current}\cos(\beta - \psi) \tag{4.26}$$

$$\dot{y}_{current} = V_{current}\sin(\beta - \psi) \tag{4.27}$$

where $x_{current}$ and $y_{current}$ are with respect to earth fixed reference frame. The above equations can be directly integrated to estimate the position coordinates which can then be added to the actual vehicle coordinates to simulate the effect of currents.

4.7 Concluding Remarks

This chapter elaborates two different and practical ways of underwater vehicle modelling. The conventional way is to study the physics of the system and build a model around it. Some generic equations are presented that can be used to model all underwater vehicles. However this process is laborious and requires experimental data to determine the hydrodynamic coefficients which is not always available. An alternate route using SI is suggested that can exploit the data collected from the vehicle during free running experiments. Interesting characteristics about system dynamics have been explored that aided in developing the model. Finally SI on *Hammerhead* data was applied and models for both rudder-yaw and hydroplane-depth channel were estimated. Problems encountered during the SI experiments were also explained specifically in the depth channel identification trials. Moreover, some cross coupling effects were observed in the depth and heading channels but were ignored for simplicity. In the forthcoming chapters, guidance systems specifically for cable tracking mission and controllers are designed for the models developed herein. It will be shown how the controllers cope with model uncertainties and external disturbances. This will be preceded by simulation results and then followed by some actual in water controller experimental results.

Chapter 5

Guidance System

The *Hammerhead* vehicle is a torpedo shaped vessel with very limited manoeuvribility. Missions such as cable following, object tracking, dams inspection and site surveying are all well suited to this class of vehicle where precise motion and hovering requirement is not an issue. Thus one of the main mission objectives set for *Hammerhead* is to track a cable/pipeline for inspection purpose. This chapter highlights two guidance laws which are designed to perform the said objective. The pure pursuit guidance is an extension of the work by Ahmad *et al.* (2003) and is employed here together with an onboard sonar for cable detection and following. A hybrid guidance law (Naeem *et al.*, 2003*a*) is also developed and explored which exploits and combines various features from several guidance systems for airborne vehicles. Simulation results are shown to demonstrate the trajectory planning capabilities of the proposed guidance strategies.

5.1 Pure Pursuit Guidance

A variety of methodologies and concepts have been devised to perform object tracking by an underwater vehicle. An account of various AUV guidance schemes has been documented by Naeem *et al.* (2003*d*) and is reported in Chapter 2. Cable tracking missions generally make use of an onboard vision system for detection purposes. The coordinates of the cable can be evaluated based on the position of the object in the field of view of the camera. For electrical or telecommunications cable, an onboard magnetometer can also be utilised for detection. The contemporary method to detect linear subsea objects is through active magnetic, passive magnetic or electromagnetic detectors mounted on an underwater vehicle (Bjerrum and Slater, 2001). These sensors provide lateral and longitudinal displacement of the vehicle from the target

pipeline, but no target direction. An additional sensor is needed to measure the target orientation. This information is then used by the autopilot to steer the vehicle over the pipeline. The work herein assumes an onboard sonar that provides target orientation and direction. A modified PNG law is proposed for tracking underwater cables/pipelines (Naeem *et al.*, 2003*c*; Naeem *et al.*, 2004*b*) which is an enhancement of the work undertaken by Ahmad *et al.* (2003).

The intent is to demonstrate the suitability of the proposed guidance scheme for detecting and tracking an undersea object, in this instance a pipeline, via simulation. The tracking of a pipeline by an AUV is first posed as an AUV-target interception problem. The classical PNG law is employed to generate the guidance command signals to the AUV. Subsequently this is modified to achieve the desired target tracking trajectory objective. A brief summary of the advantages of the use of sonar over other methods is as follows.

5.1.1 Sonars

Recent advances in sonar technology provides a sophisticated means of finding fibre optic cable, plastic, metal and other materials suspended in mid-ocean or buried in a sea bed (Bannon, 1998). This strategy entails use of an *active* sonar system for target (pipeline) detection. Active sonars employ echo ranging to detect an object whereas passive sonars pick-up acoustic radiation of ships, submarines etc, by an array of hydrophones. Some of the several other factors that influence this choice are:

1. Active sonars echo-range and therefore are capable of detecting even a submerged pipeline in the background of clutter i.e., reverberations, in which it appears. Vision based systems will have severe limitations in such a scenario which is very likely to occur at seabed due to underwater current and various other natural disturbances.

2. They can provide both range and orientation of the target, unlike magnetometers, which are non-directional and can easily mislead the AUV in presence of subsea ferrous deposits.

3. An onboard active sonar can also be employed for retrieval of an AUV back to the mother ship once mission is accomplished. This has been investigated by Ahmad *et al.* (2003) and is an area of ongoing research.

4. Sonic signals are the only practical and efficient way of long range undersea communication, for instance between the mother ship and the AUV (Whitcomb, 2000).

5.1.2 Problem definition

The following assumptions are made in order to formulate the guidance problem:

1. The AUV-target engagement is planar i.e. in the same plane.
2. Although the pipeline is a continuous object, it is convenient to assume it as a point mass moving with a constant velocity. This condition can be ensured by considering only the latest value of echoed ping received by an onboard AUV sonar. The AUV is also considered as a constant velocity point mass moving with a certain velocity.
3. Complete navigational information of the target is available to the AUV.

Consider a two-dimensional engagement geometry in which the AUV and target are closing on each other at constant velocities V_p and V_e respectively as shown in Figure 5.1. The imaginary line joining the AUV and target is the LOS which forms an angle λ with the fixed reference. From the geometry of the figure the LOS angle is given by,

$$\lambda = \tan^{-1}\frac{h}{r} \tag{5.1}$$

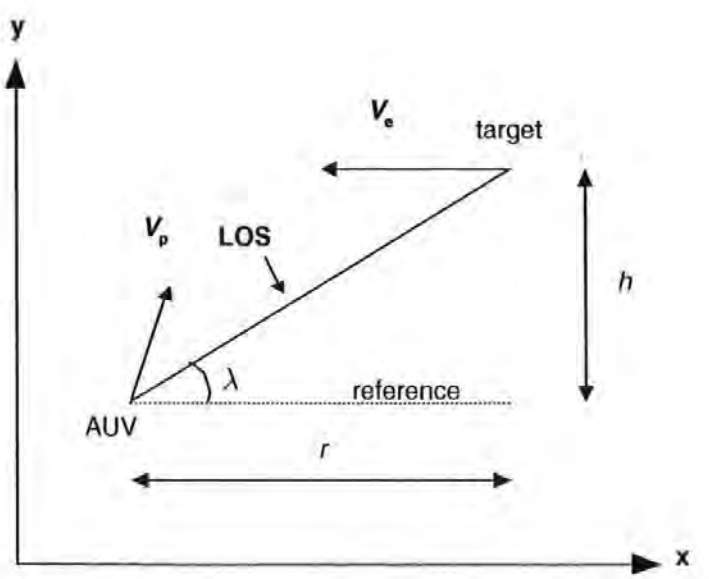


Figure 5.1: AUV-target engagement geometry

where, h and r are the relative separation between the AUV and target perpendicular and parallel to the fixed reference respectively. The relative movement between the AUV and target causes the LOS to rotate through a small angle λ, indicating a displacement h between AUV and target perpendicular to the fixed reference. The length of LOS is a range R_{LOS} and represents the initial AUV-target distance. The problem is then to develop a guidance system that will make the initial range R_{LOS} between the AUV and target as small as possible at the end of expected intercept time. It will be shown later in simulations that it is a good starting point for achieving the desired tracking objective, without actually intercepting the target.

5.1.3 Proportional navigation guidance law

The input to the guidance system are the sensor measurements. Information from the sensors is fused together and provided to the guidance system, which calculates the trajectory to be followed by the AUV. The objective of the guidance law is to generate commands for the control system to steer the AUV so that it will chase a target using a constant AUV velocity V_p and a controllable heading angle ψ_p. For cable following, the first stage of the tracking problem will be regarded as an AUV-target interception problem which is subsequently modified to realise the desired *tail-chase* type AUV trajectory. The *tail-chase* type trajectory of interest is akin to that formed when a dog is chasing a cat. This type of trajectory will ensure that the AUV is always trailing behind the target and thus continuously monitor it at a close length. From the discussion of Section 5.1.2, it is intuitive that if the AUV is made to lie on the LOS and hold it there as well, a constant relative bearing between the AUV and target is ensured that is, the LOS of sight does not rotate, and interception will occur. This mechanisation can be realised using a PNG law.

Proportional navigation is a method of guidance, which generates command signals u_c, proportional to the LOS angle λ, so that the pursuing vehicle remains on the LOS. This can be mathematically stated as:

$$u_c \propto \lambda \tag{5.2}$$

$$u_c = k\lambda \tag{5.3}$$

Where, k is called the navigation constant and is an important design parameter. A judicious choice of k will ensure that the LOS does not rotate and hence no further input command is required. Thus, it influences both, the engagement trajectory as

well as the command input. The PNG scheme is illustrated in Figure 2.7 and a good description on PNG can be found in Garnell (1980).

5.1.4 Guidance law application

For implementing the guidance law of Equation 5.3, it is necessary to compute the LOS angle λ. This requires relative positions of the AUV and target in both the coordinates i.e.,

$$h = y_e - y_p \tag{5.4}$$

$$r = x_e - x_p \tag{5.5}$$

where $(x_p, y_p), (x_e, y_e)$ are the AUV and target positions respectively in earth fixed reference frame. Hence,

$$\lambda = \tan^{-1}\left(\frac{y_e - y_p}{x_e - x_p}\right) \tag{5.6}$$

The components of the AUV velocity in the (x, y) plane can be stated as,

$$V_x = V_p \cos(\psi_p) \tag{5.7}$$

$$V_y = V_p \sin(\psi_p) \tag{5.8}$$

Hence, the differential equation for the components of the AUV position can be expressed as:

$$\dot{x}_p = V_x \tag{5.9}$$

$$\dot{y}_p = V_y \tag{5.10}$$

It is assumed that the AUV speed V_p and heading angle ψ_p are available to the guidance logic from an onboard speed log and gyro compass respectively. In certain cases both components of the AUV speed i.e., Equations 5.9 and 5.10 can be obtained directly from a Doppler log.

By integrating the above velocity component equations, the AUV position coordinates (x_p, y_p) in the earth fixed reference frame can be found. Integrating Equations 5.9 and 5.10 from time $t = 0$ to $t = t_f$, and zero initial condition, that is, $x_p(0) = 0$ and

and $y_p(0) = 0$ will give:

$$x_p = \int_0^{t_f} V_p \cos\psi_p dt \tag{5.11}$$

$$y_p = \int_0^{t_f} V_p \sin\psi_p dt \tag{5.12}$$

where t_f is time until intercept.

Similarly, it is simple to evaluate the target positions (x_e, y_e) in the earth-fixed frame. It is assumed that the target velocity V_e and orientation ψ_e is known as a function of time. These quantities can be either measured or estimated. Therefore, target positions are given by

$$x_e = x_e(0) + \int_0^{t_f} V_e \cos\psi_e dt \tag{5.13}$$

$$y_e = y_e(0) + \int_0^{t_f} V_e \sin\psi_e dt \tag{5.14}$$

Thus, by substituting Equations 5.11-5.14, in Equation 5.6, the LOS angle λ can be determined which on substitution in Equation 5.3 would generate appropriate guidance commands. This completes the guidance law mechanisation. The next section present simulation results to assess the performance of the proposed guidance system.

5.1.5 Simulation results

The ultimate objective of the work presented herein is the development and simulation of a guidance system for an AUV to follow a subsea cable/pipeline for inspection purpose. The guidance law is developed in the Matlab/Simulink environment and is shown in Figure 5.2. For guidance system simulation, the blocks denoted by "theta1" and "auv_heading" are hooked up together thus completing the guidance loop. It was mentioned in Section 5.1.2 that the AUV and pipeline are considered as point masses moving with constant velocities. It is also assumed that the AUV and target are moving at the same speed i.e.,

$$\frac{V_p}{V_e} = 1 \tag{5.15}$$

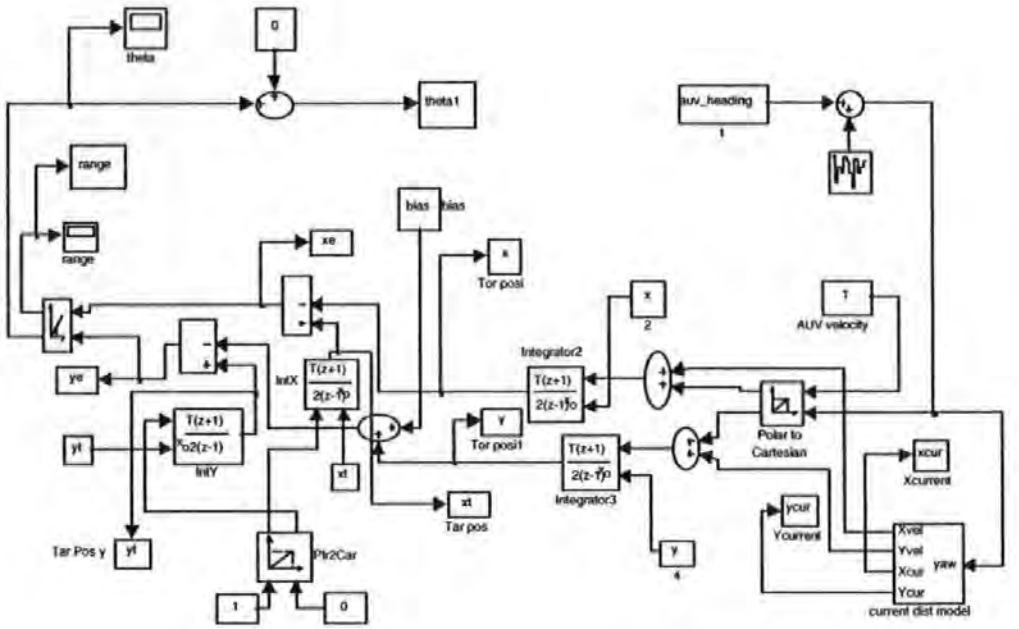


Figure 5.2: Pure pursuit guidance system block diagram in Matlab/Simulink

The target frame of coordinate (FOC) with respect to the AUV is $(0, 10)$ representing the seabed whereas, the initial AUV coordinates in the inertial 2-dimensional frame of reference (x, y) plane are $(0, 200)$ with respect to the target FOC. Further, it is assumed that the target obeys Equation 5.16, and is heading eastwards from the initial FOC.

$$\begin{aligned} x(t) &= x_0 + V_e t \\ y(t) &= h \end{aligned} \tag{5.16}$$

V_e and h begin fixed. The target, a "fleeing" pipeline is travelling at a constant distance h from the AUV's inertial FOC. The AUV is to be launched from a mother ship in the vicinity $(0, 200)$ of the target to intercept it. This completes the *tail-chase* problem definition. A navigation constant of $k = 1$ has been chosen since for this value, the AUV trajectory changes at the same rate as the imaginary LOS joining the target and the AUV. This type of flight profile is often referred as "pursuit course" and the corresponding guidance law as *pure pursuit*. The trajectory is similar to that formed by a predator when pursuing a prey, for instance, a dog-cat or hound-hare pursuit. The predator always prefers to tail chase a target rather than intercept it by establishing a lead angle. This characteristic is exploited herein to achieve the

pipeline-tracking objective and is discussed below.

Since it is desired to follow rather than intercept the target, a bias is introduced to Equation 5.4, which in effect alters the guidance signal issued by the PNG law of Equation 5.3. This essentially prevents the value of h in Equation 5.4 from reducing to zero thus precludes the AUV from intercepting the target. The value and time of introduction of the bias would be user defined, depending on at what depth above the target (pipeline) the AUV is expected to operate. In this case, a bias of 10 metres is introduced after 40 sample times of the simulation run. In a real system, this value could be introduced by a pressure-depth sensor or altimeter on an AUV, after descending to a depth of 10 meters above the seabed. The simulation is run for 300 samples and the result is depicted in Figure 5.3. The AUV heading generated by the guidance system is also shown in Figure 5.4. The AUV charts out a pursuit course for the first 150 metres of distance travelled. With the introduction of a 10 metre bias signal at the end of 150 metres, the vehicle maintains a desired longitudinal position h, while tracking the cable laterally without ever intercepting it. The transient behaviour in Figure 5.4 at approximately 40 sample times is due to the addition of the biasing signal which alters the heading angle to be followed by the AUV for the rest of the mission duration.

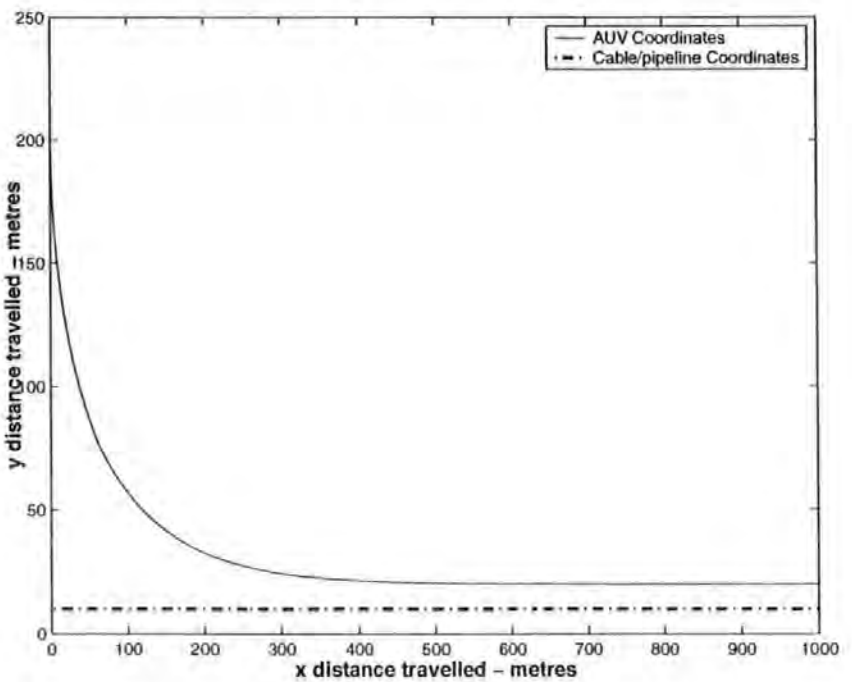


Figure 5.3: Coordinates of AUV and pipeline generated by the pure pursuit guidance system

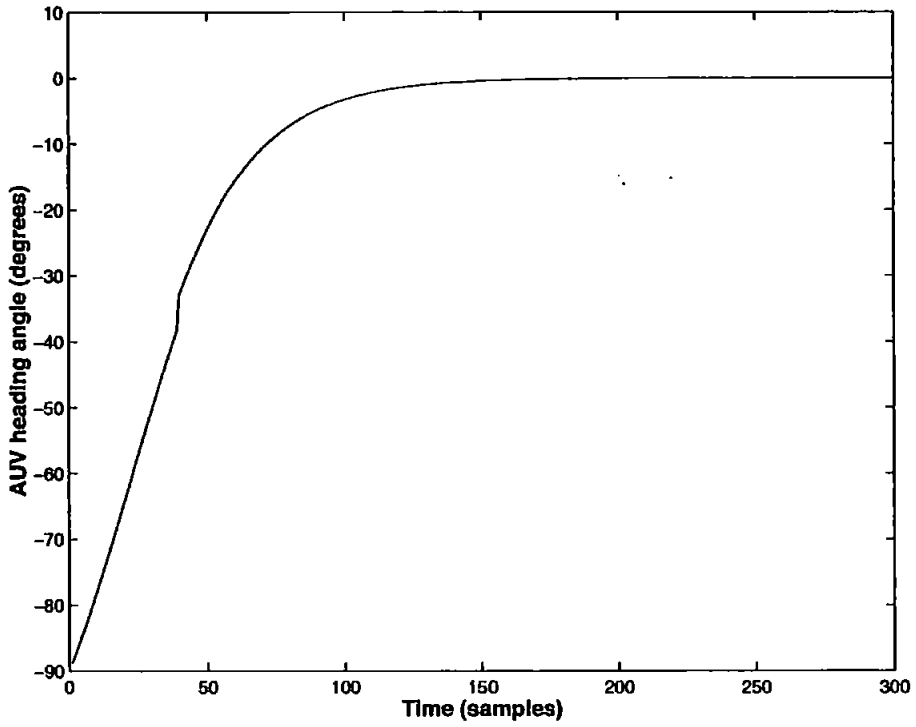


Figure 5.4: Heading angle to be followed for cable tracking

5.2 A Hybrid Guidance Law

From Chapter 2, it is evident that LOS is the key element of all guidance systems and therefore a guidance law could not be realised without employing LOS. The previous section proposed a new guidance strategy for cable tracking which is a derivative of the basic PNG law of airborne systems. This section presents another guidance scheme for underwater vehicles combining several features of the airborne and underwater vehicle guidance systems. To re-emphasise, the objective of any guidance law is to steer the AUV so that it intercepts the target in minimum time and with maximum accuracy. Fundamentally, the guidance law presented herein utilises AUV speed as a means to formulate the problem. The vehicle is launched from a mothership at its top speed which is gradually decreased when in close proximity of the target. Clearly this requires a nonlinear controller or several linear controllers designed at various speeds that covers the entire range of operating conditions. Behaviourial studies of aquatic animals reveal that they bear similar switching characteristics when in search of prey. An animal may show variations in speed or direction that are characteristic of a specific phase of tracking behaviour during the course of a successful track. For instance, lobsters show an initial increase in speed, followed by a steady speed phase,

and a final decrease, as they progress towards the forage or source of a chemical discharge (Grasso, 2001). This is similar to the AUV guidance scheme presented in this section.

5.2.1 Problem formulation

In order to formulate the hybrid guidance law, the following assumptions are taken:

i. The AUV and target are in the same plane

ii. Complete navigational information is available through onboard sensors

iii. A complete knowledge of the target's motion is available to the AUV

iv. The AUV is equipped with a vision system that generates the coordinates of the points on the cable to be tracked

v. The initial target coordinates (one end of the cable) are known prior to the mission

The first assumption of a two-dimensional engagement is taken for simplicity, however, the concept can easily be extended to multi-dimensional geometry. Moreover, the AUV is assumed to be equipped with all necessary navigational sensors which can estimate the vehicle's velocity, orientation and direction. The third and fourth assumptions are related and assumes the target (cable) as a point mass moving with a constant velocity whose coordinates can be derived using an onboard vision system. The last assumption provides the desired initial orientation (LOS angle) of the vehicle.

For the hybrid guidance system, the complete mission scenario is classified into four different phases mainly derived from airborne systems which utilises various guidance laws as shown in Figure 5.5 and are explained below

Launch Phase In the first phase called the *launch phase* or boost phase, the vehicle is launched from a submarine or a mothership in the vicinity of the target with an arbitrary orientation and is guided in the direction of the LOS with maximum speed. This phase of the mission requires the AUV to lock on to the LOS angle λ where the LOS is measured with respect to the known end of the cable. Thus the guidance law employed here is a simple LOS guidance given by Equation 5.1. Please note that a two-dimensional geometry is considered here for simplicity.

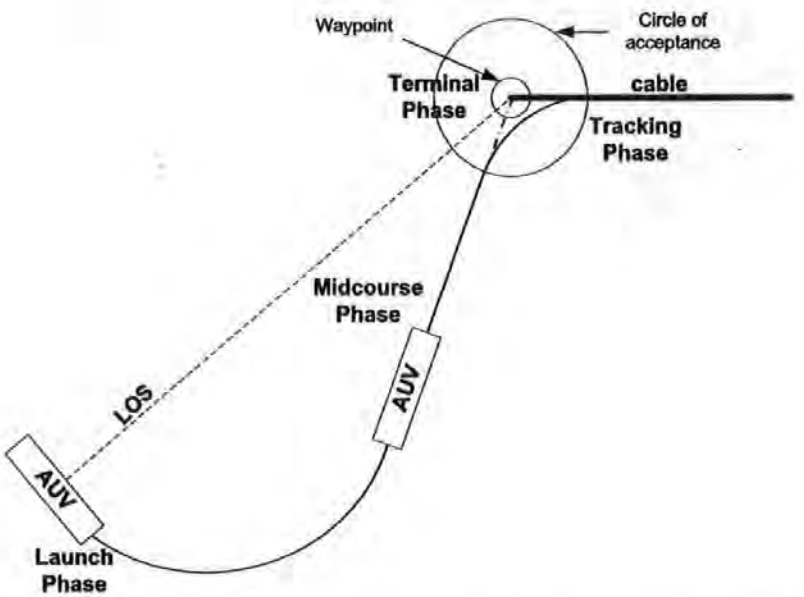


Figure 5.5: Planar view of the four phases of flight for cable tracking problem of an AUV

Midcourse Phase In the second part of the mission also called the *midcourse phase*, the vehicle follows the LOS as closely as possible. In the *midcourse phase*, the vehicle follows the LOS angle with maximum speed using waypoint guidance, (Healey and Lienard, 1993). During this part of the flight, changes may be required to bring the vehicle on to the desired course and to make certain that it stays on that course. The midcourse guidance system places the vehicle near the target area, where the system to be used in the final phase of guidance can take over. It should be noted that there is no need for the vehicle to submerge at this stage (except for surveillance operations), as the objective is to approach the target area with maximum accuracy regardless of the orientation of the vehicle with respect to the cable. Staying on the surface means that the vehicle can get GPS fixes throughout thereby reducing the probability of missing the target.

Terminal Phase When the vehicle reaches within the COA, the third phase called the *terminal phase* is invoked. During this phase the vehicle must be slowed down and submerged in order to line up with the cable/pipeline as shown in Figure 5.5. The COA in this case as opposed to Healey and Lienard (1993), should be taken at least the minimum turning radius of the vehicle in order to avoid an overshoot. At the end of the terminal phase, the vehicle must have

detected the cable in order for the tracking phase to take over. However, if the cable was passed undetected by the onboard sensors whilst the AUV made contact with the known coordinates of the cable successfully, the entire terminal phase has to be repeated and the AUV is directed again towards the known end of the cable.

Tracking Phase Finally, when the vehicle enters the waypoint, the fourth phase called the *tracking phase* is called up utilising any existing guidance law with the vehicle speed reduced to its minimum value. If the vehicle is equipped with a vision system, the cable coordinates can be evaluated by transforming the cable position in the field of view of the camera to earth coordinates. The vehicle coordinates are available through dead reckoning using an inertial navigation system. Thus the AUV can track the established LOS angle between the current AUV and cable positions with a bias in the vehicle y-position to avoid an intercept. On the other hand, if the cable to be followed is an electrical/communication cable, then magnetometers could be used to detect the radiation from the cable and guide the vehicle in the appropriate direction (Naeem *et al.*, 2003*d*).

After acquiring the necessary information from the cable inspection mission, the AUV returns to dock at its launching position for recharging and data downloading. The vehicle can then be reprogrammed for a new mission.

5.2.2 Simulation results

Since the hybrid guidance law presented exploits several vehicle speeds during different phases of the mission, therefore it is not applicable to *Hammerhead* at present. This is because the *Hammerhead* model is only available at one speed and is an area of further research. Thus a model has been chosen from literature to demonstrate the suitability of the proposed algorithm and will be applied to the *Hammerhead* in the future. The results presented in this chapter only depict the trajectory generated by the guidance system without considering any dynamics. However, the performance of the guidance system can truly be evaluated when vehicle dynamics at various speeds are taken into consideration and is the topic of Chapter 6.

To implement the guidance law, it is necessary to compute the LOS angle λ given by Equation 5.1. In addition to the LOS angle from the vehicle to the target, the guidance system also generates the range (distance) of the AUV from the target. The

range measure is used to switch between different pre-tuned controllers. The guidance system is developed in Matlab/Simulink environment and is depicted in Figure 5.6. As shown, the input to the guidance block is the vehicle speed and heading. The guidance system generates the range R and LOS angle λ which must be followed to track the cable. The guidance block is further elaborated in Figure 5.7 where the vehicle x and y coordinates are compared with the target coordinates in earth-fixed frame of reference to estimate the LOS angle and range measure.

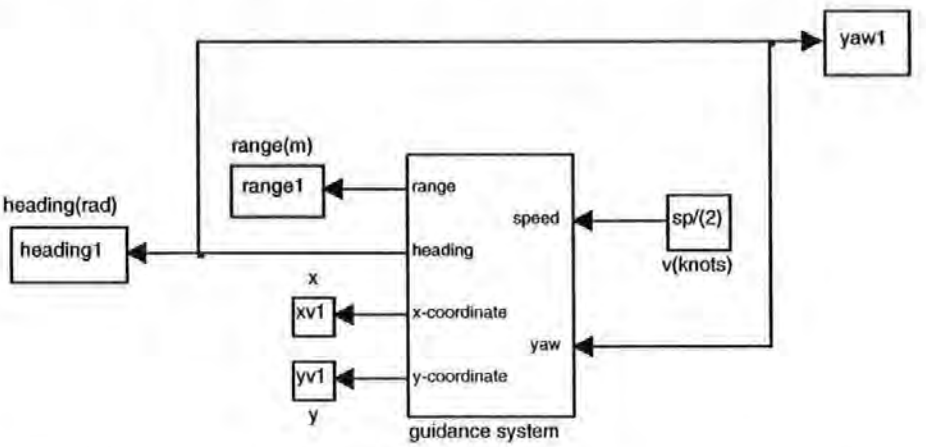


Figure 5.6: Hybrid guidance system block diagram in Matlab/Simulink

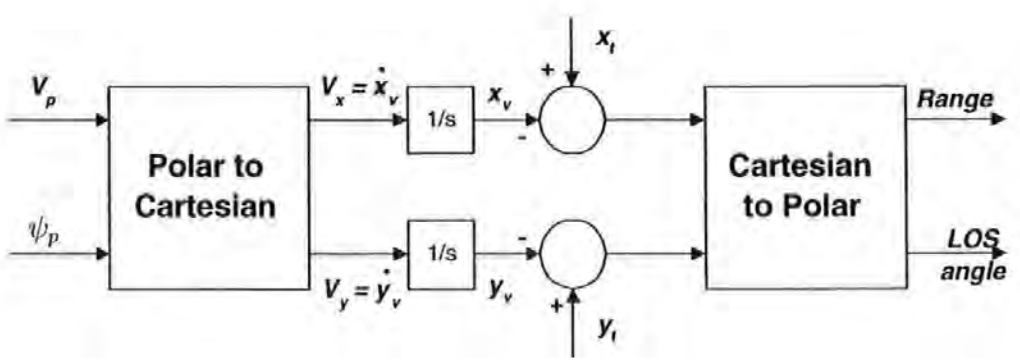


Figure 5.7: Hybrid guidance system block diagram

In the end, Figure 5.8 present simulation results of the proposed guidance system. The vehicle was launched in the vicinity of $(100, 100)$ representing the docking point. The known end of the cable was at $(200, 50)$ in earth coordinates. The vehicle initially followed the LOS at its maximum velocity and switched to a moderate speed when

within the COA, which in this case was about 25 metres representing the minimum turning radius of the vehicle. Finally, when the vehicle entered the waypoint, the radius of which was adjusted heuristically, the AUV slowed down once more for the final tracking phase to commence. The onboard vision system can then take over to follow the cable. The trajectory generated by the guidance system as shown is closely following the cable without intercepting it. It will be shown in Chapter 6 that if the vehicle speed is kept constant, this can produce an overshoot on turning and the trajectory will not be a smooth one. Figure 5.9 depicts the heading angle to be followed by the vehicle that has been generated by the hybrid guidance system. The initial heading is arbitrary while the heading angle generated during the midcourse phase is the LOS angle which is fixed at -26.56 degrees for the duration of the midcourse guidance phase. The heading trajectory then changes during the final terminal and tracking phases. Since the cable was assumed to be laid eastwards therefore the final heading angle to be followed is 0 degrees.

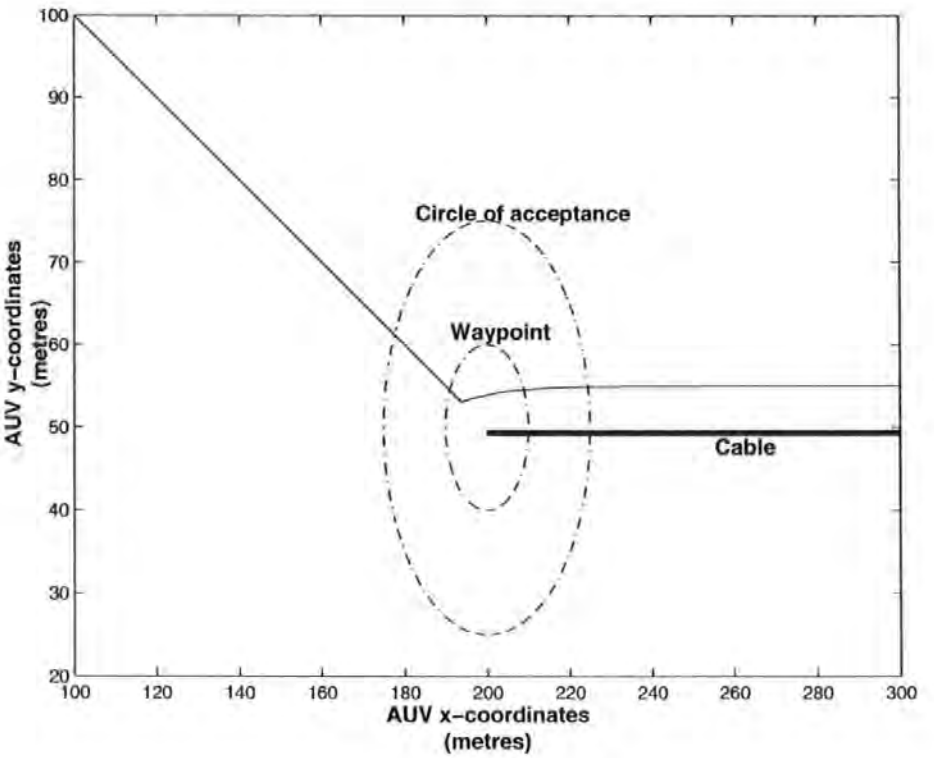


Figure 5.8: XY Trajectory generated by the hybrid guidance system

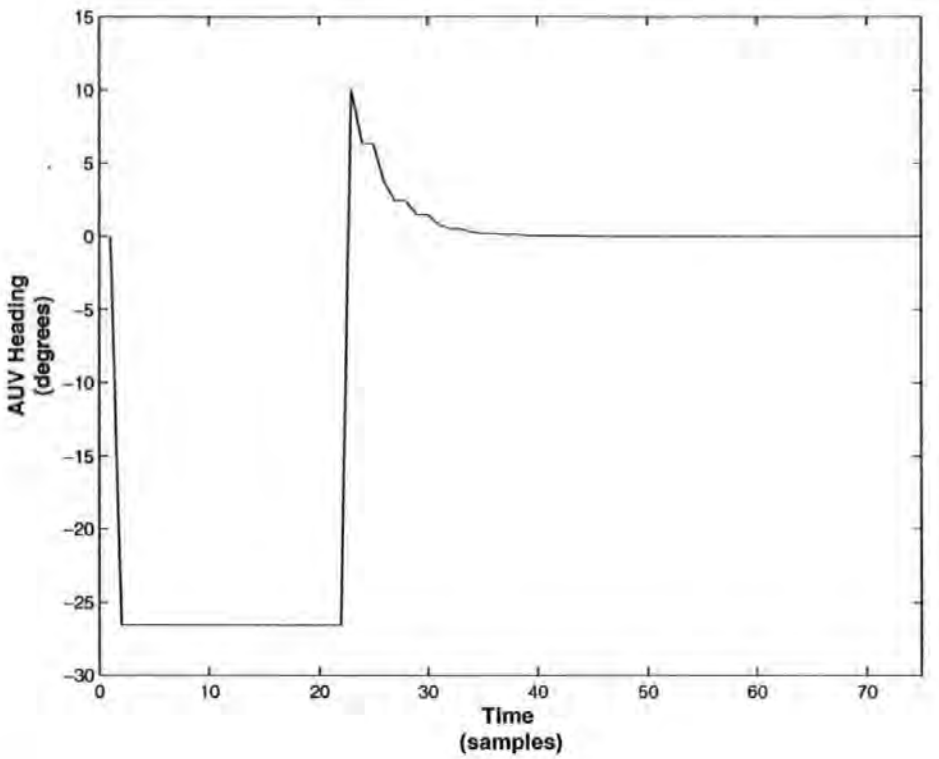


Figure 5.9: Heading angle generated by the the hybrid guidance system to be followed by the AUV

5.3 Concluding Remarks

Guidance is imperative for any autonomous vehicle for successful target interception. This chapter presents two guidance laws for the cable tracking problem. The pure pursuit guidance is a derivative of PNG for airborne missiles while the hybrid law is a switching guidance technique exploiting several existing methodologies from the literature. The guidance system simulation models have been developed in the Matlab/Simulink environment and the relevant Simulink blocks are also shown. Finally, simulation results are presented to demonstrate the trajectory planning capability of the proposed guidance systems. This will subsequently be applied to the *Hammerhead* AUV to gauge their performance. It must be pointed out here that the broader aim of the work presented is to render an underwater vehicle truly autonomous, incorporating features such as smart launch, midcourse guidance, target tracking, area search and finally, return and dock to the mother ship autonomously on completion of a given task. The next chapters aim at developing control laws for *Hammerhead* AUV along with the integration of the proposed guidance and control systems.

Chapter 6

Linear Quadratic Gaussian Controller with Loop Transfer Recovery

Development of an autopilot for the *Hammerhead* AUV is of vital importance to the absolute design stage. This chapter is the first of two to present controller design and results as applied to the *Hammerhead* AUV models identified in Chapter 4. Herein, a linear quadratic Gaussian (LQG) controller with loop transfer recovery (LTR) is developed for *Hammerhead* to control the vehicle's horizontal and vertical motions. The guidance laws proposed in Chapter 5 are also integrated with the control system and simulation results are shown for various scenarios. A linear quadratic regulator (LQR) is developed first assuming full state availability. The shortcomings of LQR are addressed and an LQG controller is formulated by combining the LQR with a Kalman filter. Finally the robustness of the LQG controller is recovered in an LQG/LTR combination.

6.1 Introduction

So far in this thesis, the development of the *Hammerhead* AUV has been described in great depth. The hardware, software and the navigational suite are described in Chapter 3 and the trial results have been presented which were performed for SI. A great deal of *Hammerhead* yaw and depth dynamics have been explored and SISO models were extracted in Chapter 4 using SI techniques on actual vehicle data. Chapter 2 reviewed several guidance laws for underwater and aerial vehicles and

the last chapter proposed some guidance laws which are to be implemented in the *Hammerhead*. The final and most crucial requirement is to design a control system that will render the vehicle truly autonomous provided that the navigation system is accurate enough. This chapter is the first of two in a series in developing control systems for *Hammerhead* AUV. It should be noted that there is a plethora of control systems available and a comprehensive review has been undertaken by Craven *et al.* (1998). However, the selection of a particular controller for an AUV is attributed to several factors. Some of them are

- Robustness to modelling errors (plant parameter variations)
- Disturbance handling characteristics
- Set point tracking and trajectory following
- Stability characteristics
- Application to linear and nonlinear plants

Recently, control strategies based on artificial intelligence (AI) have gained considerable interest in the underwater research community. Several autopilots have been designed based on AI theory alone and also combined as hybrid systems with the existing successful control schemes to improve the closed loop performance. For instance, Kwiesielewicz *et al.* (2001) developed a PD gain scheduler based on a fuzzy expert system of an AUV and compared its performance with a PD and standard model predictive controller (MPC). Results show that the fuzzy based PD controller outperforms the conventional MPC and PD autopilot. A similar approach has been adopted by Akkizidis *et al.* (2003) based on the combination of a fuzzy logic and a PD controller. Data from actual experiments has been presented and analysed extensively.

Sutton *et al.* (2000) employed two AI techniques, namely, an adaptive network-based fuzzy inference system (ANFIS) and a simulated annealing-tuned control algorithm. A PD controller is again used as a benchmark autopilot and results demonstrate the superiority of the ANFIS approach. A multivariable neurofuzzy autopilot for an AUV has been devised by Craven *et al.* (1999) employing full nonlinear six degrees of freedom model. The controller accounts for cross coupling effects among various channels. Another neural net based nonlinear adaptive control for an AUV has been developed by Li *et al.* (2002). The algorithm employs a linearly parameterised neural

network to approximate the uncertainties of the AUV dynamics. Guo *et al.* (2003) constructed a sliding mode fuzzy control for an underwater vehicle, *AUV-HM1* developed at National Taiwan University. The controller is shown to be guaranteed stable for selected shrinking and dilating factors of the fuzzy membership functions. Experimental results are shown indicating the effectiveness of the proposed approach in dealing with modelling uncertainties, nonlinearity and environmental disturbances.

Based on the above information and controller characteristics, two robust optimal control strategies and their variants have been selected as the candidate control schemes for the *Hammerhead* vehicle. The linear quadratic Gaussian (LQG) controller with loop transfer recovery (LTR), which is developed in this chapter and the model predictive controller which has been modified to accommodate various AI techniques for improved performance. This subject will be dealt with in detail in Chapter 7. The next section highlights optimal control in general followed by an overview of LQR and and Kalman filter which serves as an observer. The discrete time LQG formulation is explained in Section 6.5 whilst the LTR is covered in Section 6.6. Finally simulation results of the application of the LQG/LTR controller to the *Hammerhead* vehicle models in the horizontal and vertical planes are presented in Section 6.7. An integrated guidance and control system is also developed based on the guidance laws proposed in Chapter 5.

6.2 Motivation of Using Optimal Control

To date optimal control theory has been extensively used to solve various control engineering problems. Especially with the advent of powerful digital computers, the computation time is curtailed to a considerable extent which is normally required to solve an online optimization problem. The optimal control is simply a minimisation or maximisation problem for which an objective function is defined that could involve different design parameters or states to optimize. A general requirement for the selection of a suitable objective function is (Burl, 1999) (1) it should accurately reflect the designer's concept of good performance and (2) the control moves should be computed with a reasonable amount of effort. The later requirement is of little significance these days due to the availability of highly sophisticated, powerful and cheap computing power.

The traditional pole placement technique works by placing the poles at designer's

chosen location to attain some design specifications such as overshoot, settling time, rise time or bandwidth. However, a major disadvantage of using this technique is that the pole locations must be worked out in advance. Moreover, the controller obtained by this method is not always optimal and a trial and error procedure is adopted until the system performance coincides with the desired specifications. Optimal control theory suggests to place the poles at points such that the resulting controller is optimal in some sense. The designer does not need to know the pole locations prior to the design. All that is required to specify is the design specifications in the form of an objective (cost) function or performance index and let the algorithm decide the best pole locations to achieve the desired response.

A substantial amount of material is available on the state feedback LQR, which is an integral part of the LQG (see for example, Franklin *et al.* (1998) and D'Azzo and Houpis (1995)), therefore only the important aspects of the controller will be outlined herein. The LQR applies to linear systems whereas the cost function optimized is quadratic hence the term "linear quadratic". The LQR assumes full state feedback which is not always available. For this purpose, a Kalman filter is employed to reconstruct all the unavailable and noisy states. The resulting controller is termed as LQG since it is optimal for processes contaminated with Gaussian noise. Augmenting a Kalman filter for state estimation seriously degrades the excellent stability margins available for an LQR controller. To recover the gain and phase margins or in other words to approximate the LQG to an LQR controller, a procedure called loop transfer recovery (LTR) is often adopted.

The LQG is a robust and optimal control methodology which has been used in solving numerous control related problems. It has gained a widespread interest to be used as an autopilot for underwater as well as aerial vehicles. For example, Brown *et al.* (1994) documents a successful design of an LQG/LTR control strategy for a non-minimum phase tail controlled missile system. The algorithm reassigns the eigenstructure of the plant including pole locations and components of the eigenvectors. An F-16 lateral autopilot based on LQG/LTR paradigm has been devised by Lin *et al.* (1997) using both feedforward and feedback controllers. The algorithm is shown to be robust to measurement noise and more suitable in practical applications. An LQG adaptive control for an underwater vehicle has been developed by Tabaii *et al.* (1994) which takes into account the parametric uncertainties. Chen and Chung (1994) presents submarine depth control using an LQG control approach. The algorithm uses a Lu-

enberger observer to estimate the disturbances due to large waves. Results report significant improvement in the vertical plane movement of the submarine near the surface of a choppy sea. LQG/LTR control of an underwater vehicle has also been reported by Juul *et al.* (1994), and Triantafyllou and Grosenbaugh (1991). However, most of these papers deal with multivariable continuous time control of a vehicle assuming that the guidance commands are available. Herein, a discrete-time LQG/LTR controller is developed which is more realistic from an implementation aspect. A simple weight selection procedure is adopted that will reduce the tuning complexity of the robust controller. The next two sections elaborate on the individual components of an LQG control problem, i.e. LQR and Kalman filter which are combined on the basis of the *separation principle.*

6.3 Linear Quadratic Regulator Design

The LQR is an optimal controller which is derived on the basis of a linear model and quadratic cost function. In its original form, the LQR forces all the states to go to zero hence the term 'regulator'. For set point tracking or trajectory following, some modifications could be made as will be discussed later in this section. The major characteristics of an LQR control scheme are given below and the block diagram of the controller is illustrated in Figure 6.1.

Excellent stability margins The LQR controller is known to have excellent stability characteristics with gain margin (GM) up to infinity and over 60^0 phase margin (PM) (Burl, 1999). This means that the controller is guaranteed stable for all magnitude of disturbances. Some important measures of the stability characteristics of a system obtained from a Bode plot are defined in Appendix B.

Optimality Unlike the pole placement approach where the designer must specify the pole locations in advance, the LQR does not require this information. Moreover, since the closed loop eigenvalues are found by minimisation of a cost function, therefore the resulting controller is optimal.

Unique analytical solution As will be shown, the LQR problem can be solved analytically and the solution is unique provided that the process is linear.

Application to multivariable systems The LQR methodology is inherently a multivariable technique therefore extension from SISO to MIMO design is rather straightforward.

Full state feedback The major disadvantage of an LQR controller is that it requires full state feedback which is often not available. This is because either there is no available sensor to measure that state or the state is too noisy. This problem is generally circumvented by employing a state estimator which will be dealt with in Section 6.4.

Controllability Another requirement of the LQR controller is that the concerned plant must be controllable. This requirement can be mathematically stated in terms of the state transfer and control input matrices.

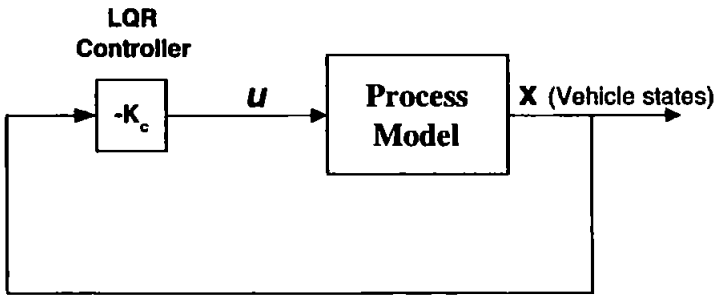


Figure 6.1: LQR control of a process

At its heart is a cost function which is the most vital element in all optimal control strategies. In LQR, a simple quadratic objective function is employed which is optimized to determine the optimum control moves and is shown below

$$J = \frac{1}{2}\sum_{k=0}^{N}\left[\mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + u^T(k)Ru(k)\right] \tag{6.1}$$

where $\mathbf{Q}$ and R are weights on process states and control input respectively. The role of $\mathbf{Q}$ and R will be further elaborated in the next subsection. The plant model in Figure 6.1 above is generally represented in state space format as given by Equation 6.2

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) \\ y(k) &= \mathbf{C}\mathbf{x}(k) \end{aligned} \tag{6.2}$$

where,

$\mathbf{x}$ n-dimensional state vector

u m-dimensional control input vector

y p-dimensional system output vector
$\mathbf{A}$ $n \times n$ matrix
$\mathbf{B}$ $n \times m$ matrix
$\mathbf{C}$ $p \times n$ matrix

A definite requirement of the LQR controller is that the plant given by Equation 6.2 must be completely *controllable* and that the matrix $\mathbf{A}$ is nonsingular. When this is true then there exists a constant feedback gain matrix $\mathbf{K}_c$ that allows the eigenvalues of the closed loop system to be assigned arbitrarily. This is mathematically stated by forming a controllability matrix $\mathbf{S}$ in terms of the matrices $\mathbf{A}$ and $\mathbf{B}$ given by

$$\mathbf{S} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A^2B} & \cdots & \mathbf{A^{n-1}B} \end{bmatrix} \tag{6.3}$$

Then the system is said to be completely state controllable if the matrix $\mathbf{S}$ has rank n. The control law of an LQR controller is then evaluated from Figure 6.1 as

$$u(k) = -\mathbf{K}_c\mathbf{x}(k) \tag{6.4}$$

It is now required to evaluate the contents of $\mathbf{K}_c$ or in other words, the state feedback gain, such that a performance index is minimised given by Equation 6.1 subject to

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) \tag{6.5}$$

where the dimension of $\mathbf{K}_c$ is $m \times n$. It should be observed that the state vector sequence $\mathbf{x}(k)$ and the input sequence $u(k)$ are not independent variables that can be arbitrarily chosen to minimise J. In this case, the minimum of Equation 6.1 is obtained by substituting zero for both $\mathbf{x}$ and u which has no physical meaning. These quantities are related through Equation 6.5 which evidently produces a nonzero value for any nonzero $\mathbf{x}(k)$ even if the input sequence is fix at zero.

Minimisation of Equation 6.1 leads to a unique solution of the state feedback gain matrix given by

$$\mathbf{K}_c = \left[R + \mathbf{B}^T\mathbf{P}\mathbf{B}\right]^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A} \tag{6.6}$$

where the matrix $\mathbf{P}$ is the positive definite solution of the discrete algebraic *Riccatti* equation shown below

$$\mathbf{P} = \mathbf{A}^T\left[\mathbf{P} - \mathbf{P}\mathbf{B}R^{-1}\mathbf{B}^T\mathbf{P}\right]\mathbf{A} + R \tag{6.7}$$

and $\mathbf{P}$ is a symmetric semi positive definite matrix i.e.,

$$\mathbf{P} = \mathbf{P}^T \geq 0 \tag{6.8}$$

given $\mathbf{P}$ and R, the problem is then to find the gain matrix, $\mathbf{K}_c$ using Equation 6.6 which on substitution in Equation 6.4 provides the required control effort.

To evaluate the eigenvalues, λ of the feedback control system, Equation 6.4 is substituted in Equation 6.2 that gives

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) - \mathbf{B}\mathbf{K}_c\mathbf{x}(k) \tag{6.9}$$

$$\mathbf{x}(k+1) = (\mathbf{A} - \mathbf{B}\mathbf{K}_c)\,\mathbf{x}(k) \tag{6.10}$$

From the above equation, the state transfer matrix of the resulting closed loop system is found to be $(\mathbf{A} - \mathbf{B}\mathbf{K}_c)$ and the characteristic equation is readily evaluated by using

$$|\lambda\mathbf{I} - (\mathbf{A} - \mathbf{B}\mathbf{K}_c)| = 0 \tag{6.11}$$

where $\mathbf{I}$ is an identity matrix of order $n \times n$. The LQR evolves a controller that is guaranteed stable and therefore all the eigenvalues of the closed loop system should lie within the unit circle.

6.3.1 Selection of weighting matrices

Control systems are often designed to specifications that involve the settling time, damping ratio and bandwidth constraint. Control systems may also be subject to constraints on the maximum output error and the maximum control input. These specifications can typically be met using the LQR after trial and error selection of the weighting matrices $\mathbf{Q}$ and R in the objective function defined in Equation 6.1. For example, the size of the control weighting matrix R can be altered until the maximum required control input, in a worst case scenario, is just under a bound imposed by the actuator. The size of the weighting matrices can also be altered to yield a desired settling time or other performance criteria. They can be used to provide tradeoff between speed of response and control effort. A higher magnitude of R with respect to $\mathbf{Q}$ reduces the amount of control effort required but at the cost of a large settling time. Conversely, a low value of R can reduce the settling time but at the expense of large control effort or even actuator saturation.

A good starting point for trial and error selection of the state weighting matrix is to set the various state contributions approximately equal. For instance, consider a system where the state consists of position and velocity of a vehicle. The magnitude of the velocity is on the order of 1/10 times the magnitude of the position in normal operation. A reasonable state weighting matrix has the weighting on the velocity 10^2 times the weighting on the position. This results in the contribution of each state being roughly equal. Initial control weighting matrix can also be selected in this manner. A simple weight selection procedure has been devised by Bryson and Ho (1975) which suggests to replace $\mathbf{Q}$ by $\mathbf{C}^T\overline{\mathbf{Q}}\mathbf{C}$ and select $\overline{\mathbf{Q}}$ and R as diagonal matrices. The elements of matrix $\overline{\mathbf{Q}}$ are assumed to be the reciprocal of the square of the maximum value of the states. Therefore if $s_{1_{max}}, s_{2_{max}}, \cdots, s_{n_{max}}$ are the maximum state contributions of an nth order process, then

$$\overline{\mathbf{Q}} = \begin{bmatrix} \frac{1}{s_{1_{max}}^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{s_{2_{max}}^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s_{n_{max}}^2} \end{bmatrix}$$

Similarly, for R, the diagonal matrix is selected as

$$R = \begin{bmatrix} \frac{1}{u_{1_{max}}^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{u_{2_{max}}^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{u_{m_{max}}^2} \end{bmatrix}$$

where it is assumed that there are m inputs and $u_{1_{max}}, u_{2_{max}}, \cdots, u_{m_{max}}$ represents the maximum allowable magnitude of the control inputs.

The final selection of weighting matrices often proceeds by trial and error, after initially incorporating all *a priori* information concerning weighting matrix selection. The designer first specifies which outputs are important to drive to zero and incorporates any physical insight available concerning to relative weighting of the terms. The feedback gains are then generated and the system is simulated and its performance is analysed. The parameters selected are adjusted again to improve the performance

which is re-evaluated and further modifications are made until an acceptable design is obtained.

6.3.2 Reference input tracking

It was mentioned earlier that the LQR is designed to regulate all states to zero. In case of set point tracking and to obtain zero steady state error, some simple modifications are required before proceeding any further. A simple way of accomplishing this has been derived by Franklin *et al.* (1998) and is depicted in Figure 6.2.

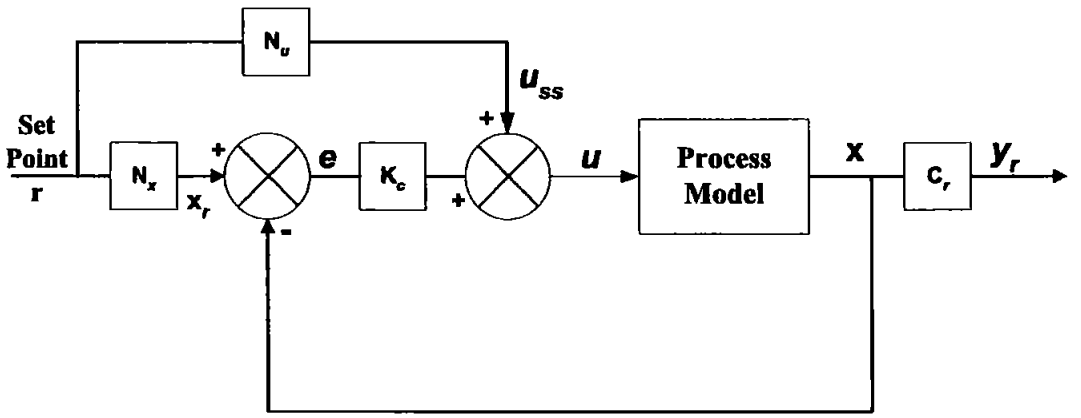


Figure 6.2: LQR control of a process for reference input tracking

The feedforward block $\mathbf{N}_u$ is used here to compensate the input for any steady state errors. However, the feedforward strategy is known to produce instability in the system. An integral control would be a better choice to get rid of any steady state output errors. This is true for type 0 systems, however, for type 1 or higher order processes, there are no steady state errors and therefore a zero magnitude will be obtained for the feedforward block. Fortunately, the *Hammerhead* yaw and depth models obtained in Chapter 4 are both type 1 and this technique would therefore provide adequate performance. The block $\mathbf{N}_x$ denotes the forward block which transforms the reference input $\mathbf{r}$ to a reference state $\mathbf{x}_r$ that is an equilibrium one for that $\mathbf{r}$. The requirement is therefore to find the contents of matrices $\mathbf{N}_u$ and $\mathbf{N}_x$. A derivation has been carried out in Appendix C and the final result is presented in Equation 6.12 below.

$$\begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{B} \\ \mathbf{C}_r & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix} \tag{6.12}$$

6.4 Kalman Filter

A serious drawback of the LQR controller is that it assumes full state availability. However, this is often not the case because (a) there are no sensors installed on the plant to measure that state due to the unavailability or cost of the measuring device, and (b) the measured states are usually corrupted with such noise levels that the required information cannot be extracted without proper treatment of the sensor readings. The Kalman filter serves both purposes by providing the estimate of unmeasured states and filters the contaminated measurements. Owing to its functionality the Kalman filter has great importance in many signal processing and control applications. In this section, the Kalman filter will be discussed as it is used in an LQG paradigm. Several text books are available on the theory and applications of Kalman filter. The reader is referred to for example Franklin *et al.* (1998) and Santina *et al.* (1994) for a comprehensive treatment on discrete time Kalman filter design.

The Kalman filter, when combined with the LQR forms a robust controller well known as LQG. It serves to provide the estimate of the unmeasured states to the LQR controller. However, the dynamics of the filter degrades the stability margins of the LQR which is recovered by constructing a fast Kalman filter. This subject will be covered in the forthcoming section on loop transfer recovery. A prerequisite of using a Kalman filter is that the system in question must be completely *observable.* This is true if every state variable of the system affects some of the outputs. This condition can be verified mathematically by forming an observability matrix $\mathbf{O}$ using the matrices $\mathbf{A}$ and $\mathbf{C}$ from Equation 6.2

$$\mathbf{O} = \begin{bmatrix} \mathbf{C} & \mathbf{CA} & \mathbf{CA}^2 & \cdots & \mathbf{CA}^{n-1} \end{bmatrix}^T \tag{6.13}$$

Then the system is said to be completely observable if the matrix $\mathbf{O}$ has rank n. In particular, if $\mathbf{O}$ is a square matrix, then the system is completely observable if the matrix $\mathbf{O}$ is nonsingular.

Suppose $\hat{\mathbf{x}}$ is a vector representing the estimate of the states produced by the Kalman filter then the state vector $\mathbf{x}$ in Equation 6.4 is replaced by $\hat{\mathbf{x}}$. In practice, $\hat{\mathbf{x}}$ will not equal $\mathbf{x}$ because the model is not perfect and there are unmodelled disturbances and the sensors have some errors and added noise. A Kalman state estimator can be constructed in the following two ways i.e.,

- Predictor estimator
- Current estimator

For systems, where the calculation time of a control move is greater than the sampling period, predictors are generally employed. A predictor estimator uses the past measurements of the state to update its estimate. On the other hand, a current estimator is employed where the computation time is low as compared to the sampling time. The current estimator provides the estimate of the states based on current output measurements. A predictor estimator would be inappropriate in this case since the delay of almost a cycle between the measurement and the proper time to apply the resulting control calculation represents an unnecessary waste. The flow chart of the Kalman filter loop is depicted in Figure 6.3 and the update equations are given below (Brown and Hwang, 1997)

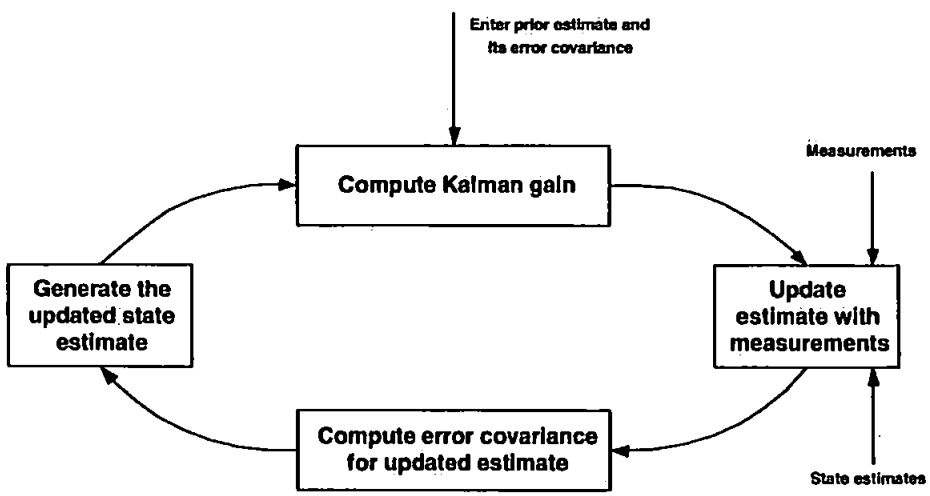


Figure 6.3: Kalman filter loop

Kalman filter gain:

$$\mathbf{K}_f(k) = \mathbf{P}^-(k)\mathbf{C}^T(k)\left(\mathbf{C}(k)\mathbf{P}^-(k)\mathbf{C}^T(k) + \mathbf{R}_v(k)\right)^{-1} \quad (6.14)$$

State estimate:

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}^-(k) + \mathbf{K}_f(k)\left(z(k) - C(k)\hat{\mathbf{x}}^-(k)\right) \quad (6.15)$$

Error covariance matrix:

$$\mathbf{P}(k) = \left(\mathbf{I} - \mathbf{K}_f(k)\mathbf{C}(k)\right)\mathbf{P}^-(k) \quad (6.16)$$

Updated state estimate:

$$\begin{aligned} \hat{\mathbf{x}}^{-}(k+1) &= \mathbf{A}(k)\hat{\mathbf{x}}(k) \\ \mathbf{P}^{-}(k+1) &= \mathbf{A}(k)\mathbf{P}(k)\mathbf{A}^{T}(k) + \mathbf{R}_{w}(k) \end{aligned} \tag{6.17}$$

where

$\mathbf{K}_f(k)$ Kalman filter gain
$\mathbf{P}(k)$ Error covariance matrix
$\mathbf{R}_v$ Measurement noise covariance matrix
and
$\mathbf{R}_w$ Process noise covariance matrix

In practice, a time varying Kalman filter requires the Kalman filter equations, the Kalman gain equations, a state space model of the plant, an initial state estimate, an initial estimation error covariance matrix, a measurement noise covariance matrix and a plant noise covariance matrix. For processes that are to operate for long periods of time, it is reasonable to use a steady state Kalman filter having a fixed Kalman gain. This prevents the computation of the Kalman gain online at each sampling interval thus saving precious processing time. This constant gain is generally used in an LQG control strategy.

Let the system be as described by the model given by Equation 6.2. The design objective is to find the Kalman gain $\mathbf{K}_f$ so that the estimate of $\mathbf{x}(k)$ is optimal. The solution to this problem is given by the discrete steady state Kalman filter gain equation given by Franklin *et al.* (1998) and Maciejowski (1985) as

$$\mathbf{K}_f = \mathbf{P}\mathbf{C}^{T}\left(\mathbf{C}\mathbf{P}\mathbf{C}^{T} + \mathbf{R}_v\right)^{-1} \tag{6.18}$$

where $\mathbf{P}$ is the steady state error covariance matrix given by the solution of a discrete steady state Riccatti equation, (Maciejowski, 1985)

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^{T} - \mathbf{A}\mathbf{P}\mathbf{C}^{T}\left(\mathbf{C}\mathbf{P}\mathbf{C}^{T} + \mathbf{R}_v\right)^{-1}\mathbf{C}\mathbf{P}\mathbf{A}^{T} + \mathbf{R}_w \tag{6.19}$$

The parameters $\mathbf{R}_w$ and $\mathbf{R}_v$ are tuned until the desired specifications are met. This is further elaborated in the following section.

6.4.1 Selection of noise covariance matrices

The parameters to evaluate the constant gain Kalman filter are the process noise and measurement noise covariance matrices $\mathbf{R}_w$ and $\mathbf{R}_v$. These are symmetric and positive semidefinite matrices which can be mathematically expressed as

$$\mathbf{R}_w = E[w(k)w^T(k)] \geq 0 \tag{6.20}$$

$$\mathbf{R}_v = E[v(k)v^T(k)] \geq 0 \tag{6.21}$$

The two noise sequences are assumed to be zero mean and uncorrelated with one another i.e.,

$$E[w(k)] = 0 \tag{6.22}$$

$$E[v(k)] = 0 \tag{6.23}$$

and

$$E[v(j)w(k)] = 0, \qquad \forall\ j \text{ and } k \tag{6.24}$$

If there is more than one process or measurement noise components, there is usually no information on the cross correlation of the noise elements and hence $\mathbf{R}_w$ and $\mathbf{R}_v$ are selected as diagonal matrices. The magnitude of the diagonal elements are taken as the variances of the individual noise components.

The effect of varying the process and measurement noise covariance matrices can be explained by considering the Kalman filter as integrated with an LQR to form an LQG controller. Keeping the state and control weighting matrices $\mathbf{Q}$ and R of the LQR controller fixed, the noise covariances are varied and the effects are examined. Suppose the process noise covariance of the plant $\mathbf{R}_w$ is increased. In this case, the Kalman gains are larger and approach steady state more rapidly, whilst the state feedback gains remain the same. The larger Kalman gains make the estimator faster, but the larger plant noise means that the estimates are less accurate. The control input also increases due to the larger gains and increased plant noise. Next the spectral density of the measurement noise has increased keeping the state feedback gain matrix constant. In this case, the Kalman gains are smaller and approach steady state more slowly. The smaller Kalman gains make the estimator slower, and the larger measurement noise makes the estimates less accurate. The error becomes larger due to the increased measurement noise, and this will translate into an increase in

the control input (Burl, 1999).

From the above discussion, it is concluded that small magnitudes of measurement noise covariance would provide an adequate performance. The process noise could then be tuned to further improve the performance of the closed loop system. Simulation results will provide some more insight on the behaviour of the Kalman filter with respect to the change in magnitude of the noise covariance matrices.

6.5 Linear Quadratic Gaussian Controller (LQG)

In the preceding sections, the LQR controller and Kalman filter have been developed, therefore attention is now focused on combining them to form what is called an LQG controller. The LQG is an optimal controller whose name is derived from the fact that it assumes a linear system, quadratic cost function and Gaussian noise. The concept of LQG, in general, is quite similar to an LQR controller which is solved assuming that all states are available for feedback. This is not always true and a Kalman filter as an observer provides an estimate of the unmeasured and noisy states. The LQR and Kalman filter are designed independently and then fused together to form an LQG controller, a fact known as the *separation principle*. An LQG controller for set point tracking is shown in Figure 6.4.

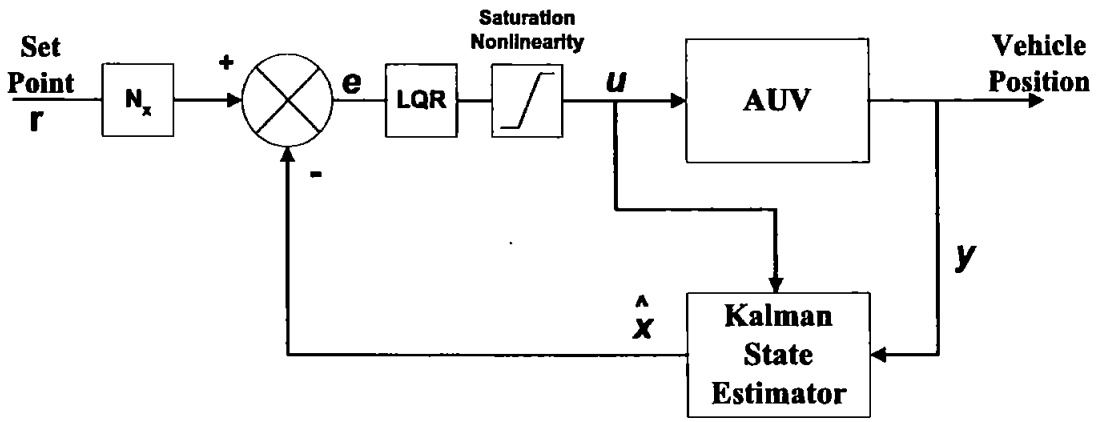


Figure 6.4: LQG controller showing LQR gain and state estimator

The state vector x in the LQR control law in Equation 6.4 is now replaced by the estimate of the states provided by the Kalman filter.

$$u(k) = -\mathbf{K}_c\hat{\mathbf{x}}(k), \qquad \text{for } \mathbf{r} = 0 \tag{6.25}$$

where $\hat{\mathbf{x}}$ is the state estimate and u is the control action. A feedback compensator is finally synthesised as a series connection of the Kalman filter and the optimal state-feedback controller given by Maciejowski (1985) as

$$\begin{aligned} \hat{\mathbf{x}}(k+1) &= (\mathbf{A} - \mathbf{B}\mathbf{K}_c - \mathbf{K}_f\mathbf{C}\mathbf{A} + \mathbf{K}_f\mathbf{C}\mathbf{B}\mathbf{K}_c)\,\hat{\mathbf{x}}(k) + (\mathbf{A} - \mathbf{B}\mathbf{K}_c)\,\mathbf{K}_f\mathbf{e}(k) \\ u(k) &= \mathbf{K}_c(\mathbf{I} - \mathbf{K}_f\mathbf{C})\,\hat{\mathbf{x}}(k) + \mathbf{K}_c\mathbf{K}_f\mathbf{e}(k) \end{aligned} \tag{6.26}$$

where $\mathbf{e} = \mathbf{r} - \hat{\mathbf{x}}$, is the error between a reference signal $\mathbf{r}$ and plant states $\hat{\mathbf{x}}$.

6.6 LQG with Loop Transfer Recovery

Loop transfer recovery (LTR) is a method of reviving the stability margins of an LQR controller which are degraded by the augmentation of a Kalman filter. The LTR technique works by making the Kalman filter sufficiently fast so that its dynamics can be ignored. In addition, by cutting the control input path to the Kalman filter, asymptotic recovery of the stability margins is achieved, given that the plant obeys some specific characteristics.

The dependence of the control input to the Kalman filter can be relaxed by adding fictitious noise to the process input. This noise can remove the dependence of the control input in addition to making the Kalman filter fast enough by effectively cancelling some of the plant zeros and possibly some of the stable poles and inserts the estimator's zeros (Maciejowski, 1985; Skogestad and Postlethwaite, 1996). One of the main limitations of LTR strategy is that it is applicable to minimum phase systems since cancelling a zero outside the unit circle produces an unstable estimator which is highly undesirable. Please note that the fictitious noise is only added during the design stage and is not actually used during the implementation.

In this section, a discrete time LQG/LTR design is presented motivated from the work of Maciejowski (1985). The Kalman gain is first evaluated by using Equations 6.18 and 6.19. The parameters $\mathbf{R}_w$ and $\mathbf{R}_v$ are then tuned until the desired filter's open-loop return ratio $\Phi(z)$ specifications are met which is shown below

$$\Phi(z) = \mathbf{C}\,(z\mathbf{I} - \mathbf{A})^{-1}\,\mathbf{A} * \mathbf{K}_f \tag{6.27}$$

Once the desired open loop specifications are met, an LQR controller is designed

based on an automatic procedure given by Maciejowski (1985). This method aims at reducing the time required to tune an LQG/LTR controller. Given the plant is minimum phase and that $\det(\mathbf{CB}) \neq 0$, the matrices $\mathbf{Q}$ and R to evaluate the state feedback gain matrix $\mathbf{K}_c$ are chosen as

$$\mathbf{Q} = \mathbf{C}^T\mathbf{C}, \text{ and} \tag{6.28}$$

$$R \approx 0 \tag{6.29}$$

These values provide asymptotic recovery of the stability margins provided that the above conditions are true. The state feedback matrix $\mathbf{K}_c$ is obtained by solving equations dual to Equations 6.18 and 6.19, and is used to generate the control according to

$$u(k) = -\mathbf{K}_c\hat{\mathbf{x}}(k) \tag{6.30}$$

A closed form solution of $\mathbf{K}_c$ for the nominal values of weighting matrices $\mathbf{Q}$ and R in Equation 6.28 and 6.29 respectively is also given by Maciejowski (1985) as

$$\mathbf{K}_c = (\mathbf{CB})^{-1}\mathbf{CA} \tag{6.31}$$

Let $\mathbf{G}(z)$ be the transfer function of the system defined by Equation 6.2 and $\mathbf{H}(z)$ be the compensator transfer function. If the plant $\mathbf{G}(z)$ is minimum phase and $det(\mathbf{CB}) \neq 0$, then full recovery is achieved if

$$\mathbf{G}(z)\mathbf{H}(z) = \Phi(z) \tag{6.32}$$

where $\mathbf{G}(z)\mathbf{H}(z)$ is called the loop transfer function.

6.7 Simulation Results

Simulation results are now presented of the development and implementation of an LQG/LTR controller to the *Hammerhead* vehicle. Several results of the application of the said controller to the yaw and depth dynamics will be explored. The *Hammerhead* yaw and depth dynamic models have been identified in Chapter 4 using SI techniques on actual AUV input-output data. The models have been represented both in transfer function and state space formats, however, the LQG/LTR requires a state space version to estimate the states and evaluate the state feedback gain matrix.

The following is a list of ingredients which are required to blend to form an LQG/LTR controller.

1. a state space model of the plant
2. a measurement noise covariance matrix $\mathbf{R}_v$
3. a process noise covariance matrix $\mathbf{R}_w$
4. a state weighting matrix $\mathbf{Q}$
5. a control input weighting matrix $\mathbf{R}$

Requirements 1, 4 and 5 are for the design of an LQR controller whereas 1, 2 and 3 are used to develop a Kalman filter. The selection of these weighting matrices having been discussed in the preceding sections.

It should be pointed out here that the *Hammerhead* onboard sensors provide bearing range of 0 to 360°. This is not an issue in simulations, however in practice, it presents severe problems as there is always a transient like effect in the output response when the vehicle crosses the magnetic north boundary from either clockwise or counter clockwise direction.

The simplest way to rectify this problem is to unwrap the heading angle through software so that the vehicle heading seems continuous to the controller. Another remedy is to measure the difference between the initial and desired yaw and the decision is made based on the minimal path length. If the shortest path suggests crossing the magnetic north, the 0 to 360° range is split up into 0 to ±180° and the AUV manoeuvres as normal following the shortest route. For instance, let the start heading angle be 20° measured with respect to the north and the desired orientation is 300° or −60°. Then the shortest path is counter clockwise from 20° to −60° (80° heading change) as compared to the clockwise path which has a span of 280° as illustrated on a polar plot in Figure 6.5.

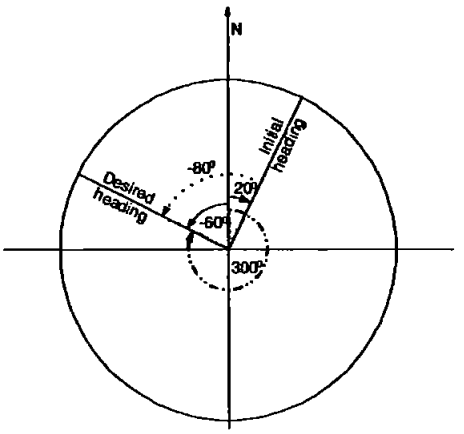


Figure 6.5: Representation of shortest path on a polar plot

The first step in an LQG/LTR control problem is to design the target filter's open-loop return ratio $\Phi(z)$ given by Equation 6.27, which requires the Kalman gain to be evaluated. By manipulating the spectral density matrices $\mathbf{R}_w$ and $\mathbf{R}_v$ in Equations 6.18 and 6.19, the Kalman filter can be designed and hence the target filter's open-loop return ratio, $\Phi(z)$. Herein, the procedure adopted by Weerasooriya and Phan (1995) is followed. In this method, the measurement noise spectral density $\mathbf{R}_v$ is kept fixed at unity and the process noise spectral density is varied until the desired frequency domain specifications are met. These specifications can be evaluated by generating the Bode plot of the open loop return ratio $\Phi(z)$ of the plant.

Two independent LQG/LTR controllers each for the *Hammerhead* yaw and depth dynamics are developed using the above mentioned methodology. For the integrated hybrid guidance and control systems simulations, since the vehicle speed plays the vital role in the guidance law formulation, therefore LQG/LTR controllers need to be developed for the models at several vehicle velocities. Please note that the *Hammerhead* model cannot be used for the hybrid guidance and control simulations since it is only available at a fix speed. A model has thus been borrowed from the literature which is represented in terms of vehicle velocity. The following generic assumptions are taken in all simulation results shown in this chapter.

- The AUV and target are in the same plane.
- Complete navigational information is available through onboard sensors.

6.7.1 Heading control

Since the heading of the AUV corrupted by noise is the only measured variable, the remaining states have to be measured through a state estimator prior to control calculations. A current estimator is used throughout since the estimate is based on the current measurement. This is because the processing time required to compute each control signal is small in contrast to the sampling time. In addition, this scheme gives more accurate results as compared to a prediction estimator (Franklin *et al.*, 1998).

Simulation results are first shown for a simple LQR state-feedback control of *Hammerhead* heading assuming that all states are available. The design of an LQG/LTR controller for *Hammerhead* yaw dynamical model is followed. The PNG law and hybrid guidance system proposed in Chapter 5 for cable tracking are then integrated with the controller and simulation results are illustrated with and without the presence of sea current disturbances.

LOS tracking using LQR

A simple LQR heading controller is shown here as an example of the effects of changing the weighting matrices on the output response. The *Hammerhead* yaw dynamic model has been used to demonstrate this phenomenon. The state weighting matrix $\mathbf{Q}$ is chosen unity while the input weighting matrix R is varied. Simulation results are depicted in Figures 6.6 and 6.7 for four different magnitudes of R. The plots clearly indicates the effect of changing R on the control input in Figure 6.7. Increasing the magnitude of R prevents excessive movements of the rudder. However, the output response (heading of the vehicle) becomes sluggish. Moreover, it is noticed that for small values of R, the rudder movement is way outside the actuator saturation limits of $\pm 22^0$. A simple remedy is to further increase the magnitude of R until the rudder movement confines just under the specified actuator bounds, however, the settling time of the heading response will become even slower. Another solution is to add a saturation block in series with the controller which will essentially clip the large rudder movements to the constrained boundaries. Herein, a saturation block is used as depicted in Figure 6.4 for the LQG/LTR controller and the constrained limits are set to $\pm 20^0$. An LQG/LTR controller will now be developed for the *Hammerhead* vehicle in the horizontal plane followed by simulation results for various situations.

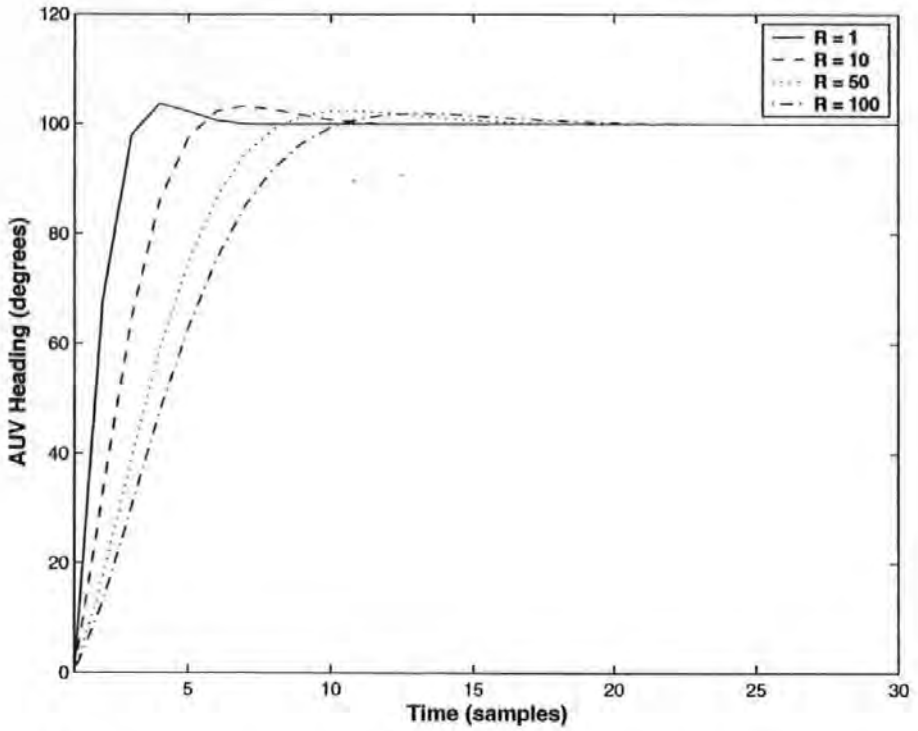


Figure 6.6: LQR heading control of *Hammerhead* showing the effects of changing R on the heading response

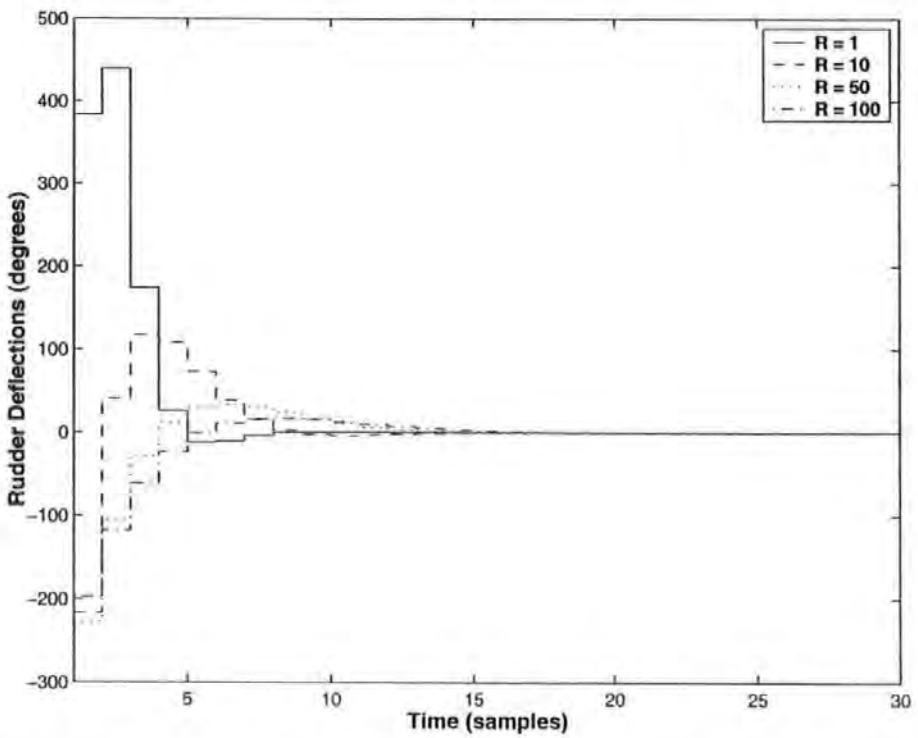


Figure 6.7: LQR heading control of *Hammerhead* showing the effects of changing R on the control input

LQG/LTR Controller Design Specifications

The LQG/LTR is a frequency domain technique and a Bode plot is normally used to determine the controller characteristics. The frequency domain parameters have been discussed previously which includes the GM and PM. An acceptable controller design usually is one that attains both a GM $\geq$ 3 dB and PM $\geq 30^0$, (Wolovich, 1994). The desired GM to achieve in this case is set at 12 dB while the PM at 58^0 which are well above the nominal values. Based on these specifications, and following the tuning methodology outlined previously, an LQG/LTR controller is designed for the *Hammerhead* yaw rudder channel. The measurement noise covariance matrix $\mathbf{R}_v$ is assumed unity while the process noise covariance matrix $\mathbf{R}_w$ is adjusted to $30\mathbf{I}_2$ (where $\mathbf{I}_2$ is a 2×2 identity matrix) to achieve the desired response. The frequency characteristics of the target's filter open loop return ratio $\mathbf{\Phi}(z)$ is plotted in Figure 6.8.

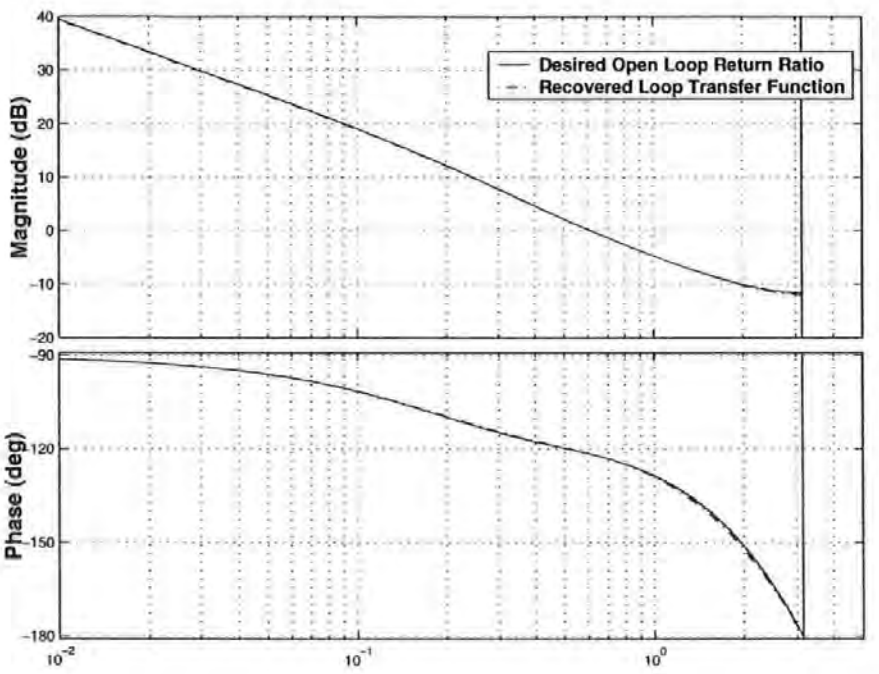
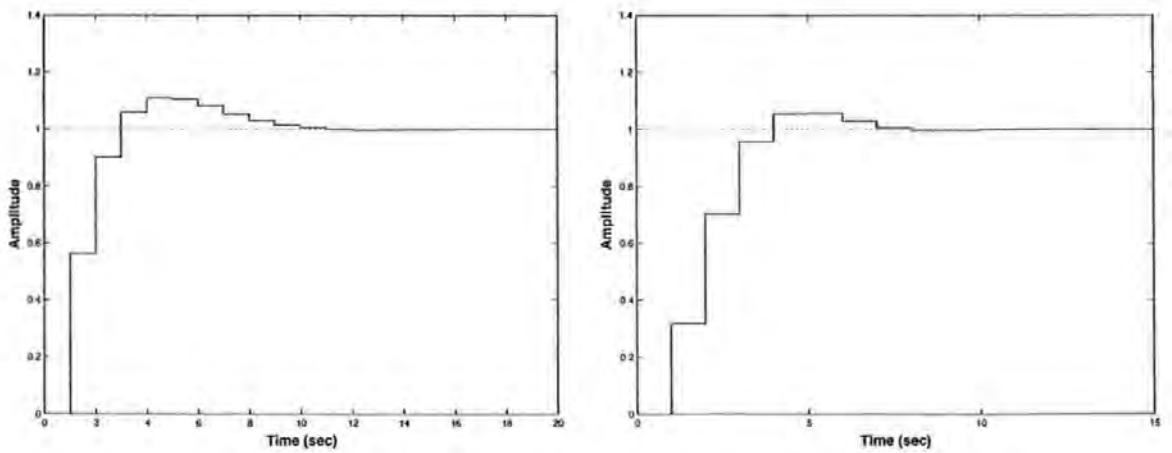


Figure 6.8: Bode plot of the open loop return ratio $\mathbf{\Phi}(z)$ (desired frequency domain specifications)

The stability margins can be readily obtained from the plot which are quite close to the desired specifications. The LQR controller is then designed using the nominal values by noting that $(\mathbf{CB}) \neq 0$ and that the plant is minimum phase. A feedback compensator is finally synthesised using Equation 6.26. The Bode plot of the loop transfer function $\mathbf{G}(z)\mathbf{H}(z)$, also shown in Figure 6.8, is superimposed on the filter's

open loop return ratio which shows excellent recovery achieved thus validating Equation 6.32. The unit step response of the closed loop system, depicted in Figure 6.9(a), bears an overshoot which needs to be minimised by increasing the weight on control input. The step response is again plotted in Figure 6.9(b) with acceptable levels of overshoot and settling time. This reduction in overshoot comes at the cost of reduced stability margins as shown in Figure 6.10. However, the stability values are deemed adequate for controller testing purpose. Figure 6.11 depicts controller block diagram constructed in Simulink where the variable "ref_heading" is a mission parameter defined by the operator or obtained from the guidance system. The variable "auv_heading" is the feedback to the controller in radians.



(a) Step response of the closed loop system with an LQG/LTR controller with nominal R

(b) Step response of the closed loop system with an LQG/LTR controller with modified R

Figure 6.9: Effect of changing R on the closed loop step response of the LQG/LTR controller

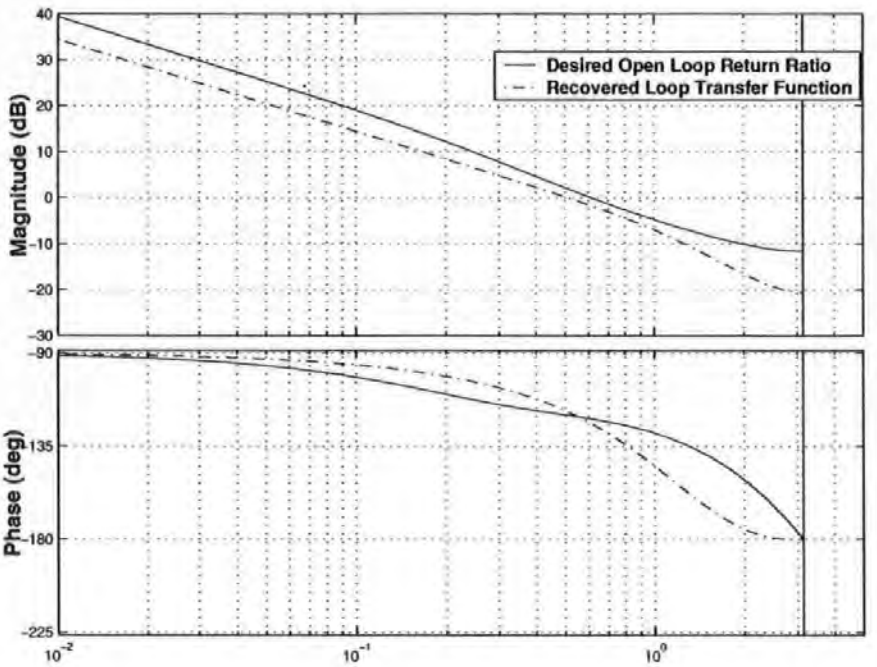


Figure 6.10: Bode plot of the loop transfer function showing degraded stability margins due to deviation from nominal values

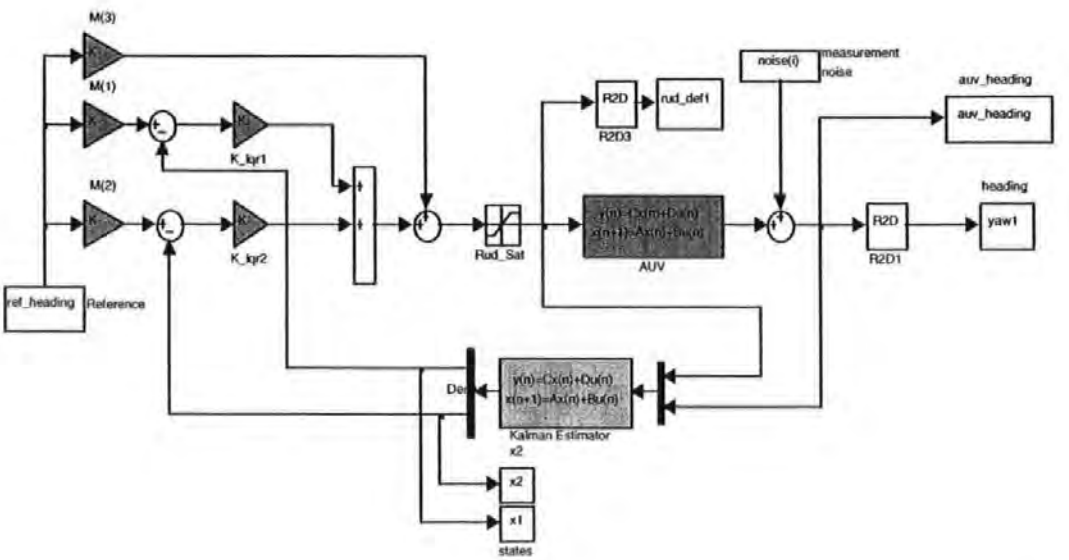


Figure 6.11: Simulink block diagram of the LQG/LTR heading controller for the *Hammerhead* AUV

The *Hammerhead* heading controller design is now complete and is next tested in simulations for various settings. First, simulation results are shown for LOS tracking. This is followed by LQG/LTR controller integrated with waypoint guidance which is essentially equivalent to multiple LOS trackings. Results will be shown with and without the presence of sea current disturbances to demonstrate the efficacy of the approach. Finally, the guidance laws proposed in Chapter 5 for cable following are integrated with the LQG/LTR based autopilot and simulation results are shown to demonstrate the cable tracking capabilities of the integrated system.

LOS following

The algorithm is first simulated to track a specified LOS angle. The vehicle is assumed to be pointing in an arbitrary direction and is required to follow a certain heading angle closely without much control effort. A saturation block is inserted in series with the controller with cutoff limits of $\pm 20^0$. A heading angle of 100^0 is chosen with the vehicle initiating close to 0^0. Nominal values are chosen for the LQR controller design whilst the measurement and process noise covariance matrices are adjusted as discussed in the preceding section to achieve the desired closed loop frequency response. The algorithm is simulated and the output response is depicted in Figure 6.12.

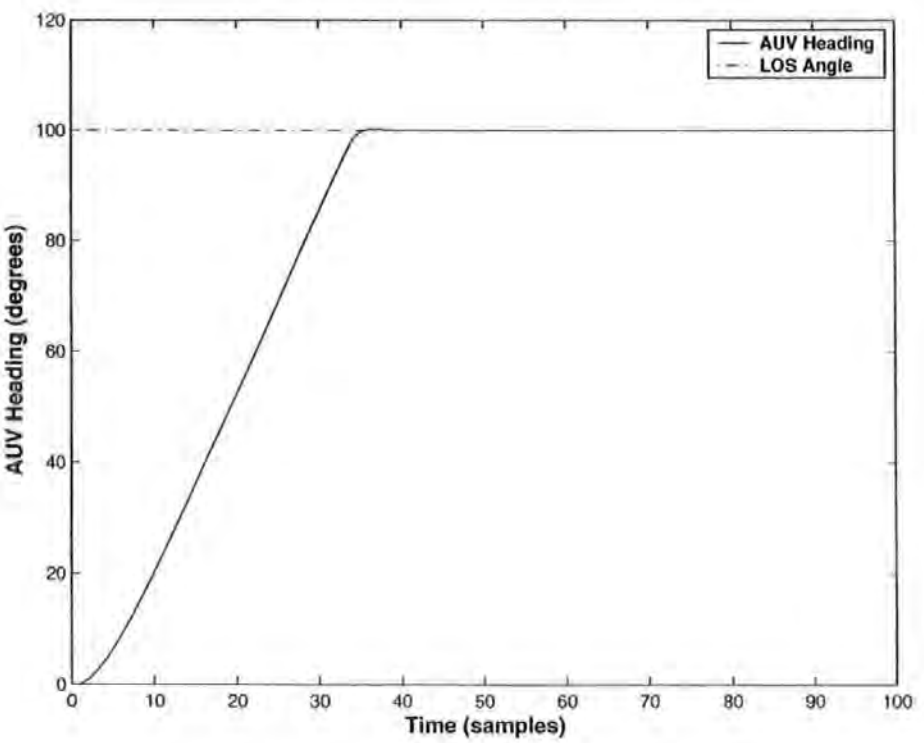


Figure 6.12: LQG/LTR control of *Hammerhead* showing LOS tracking

Similar results have also been reported in Naeem *et al.* (2003*b*). The AUV heading bears a negligible overshoot and the settling time is less than 40 samples[1]. However, the price to pay for this settling time is that the actuator saturation constraints become active for about 35 samples in the beginning of the simulation run when the vehicle was making a turn as depicted in Figure 6.13. The rudder response achieves the steady state value of 0^0 as the vehicle follows the desired heading angle of 100^0.

Increasing the control input weighting matrix R can help limit the rudder movement within the constrained boundaries and avoid this saturation but at the cost of large settling times as shown in Figure 6.14. In addition, the stability margins vary considerably. In this case, however, the GM and PM has increased but at the cost of reduced crossover frequencies of the closed loop system as shown in Figure 6.15.

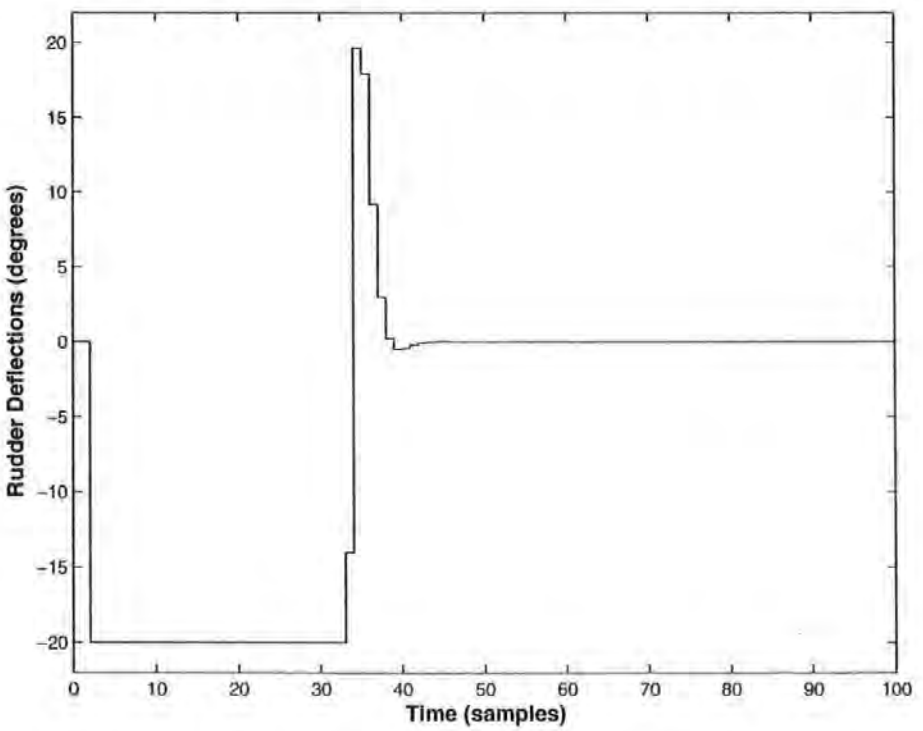


Figure 6.13: Rudder deflections generated by the LQG/LTR controller to control the *Hammerhead* heading

[1]since $Ts = 1$, therefore 1 sample time corresponds to 1 second

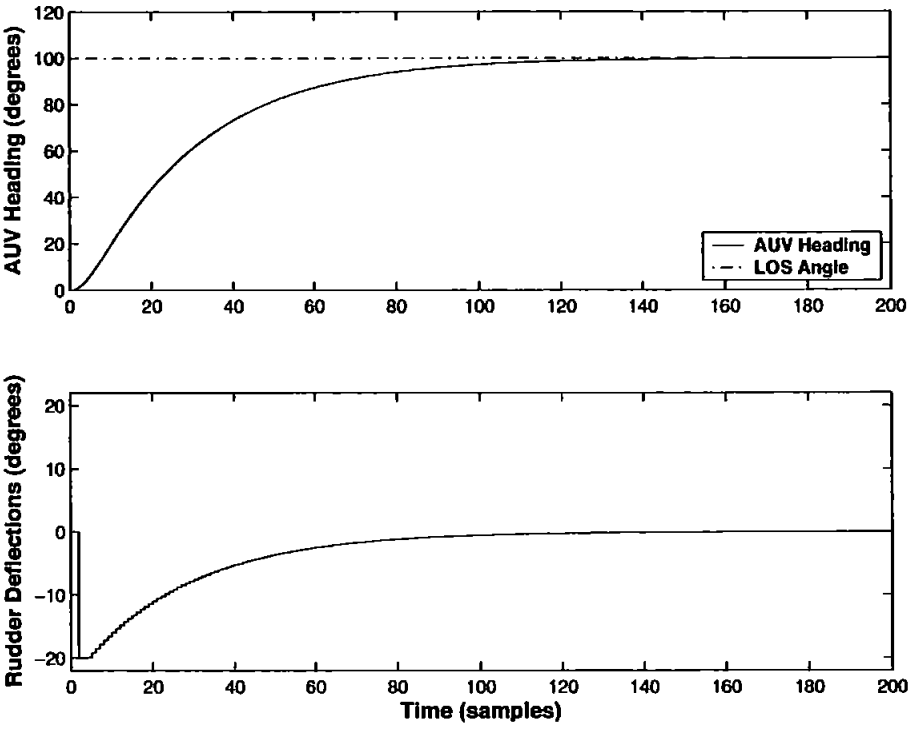


Figure 6.14: LQG/LTR control of *Hammerhead* AUV showing large settling time due to the increase in the magnitude of R

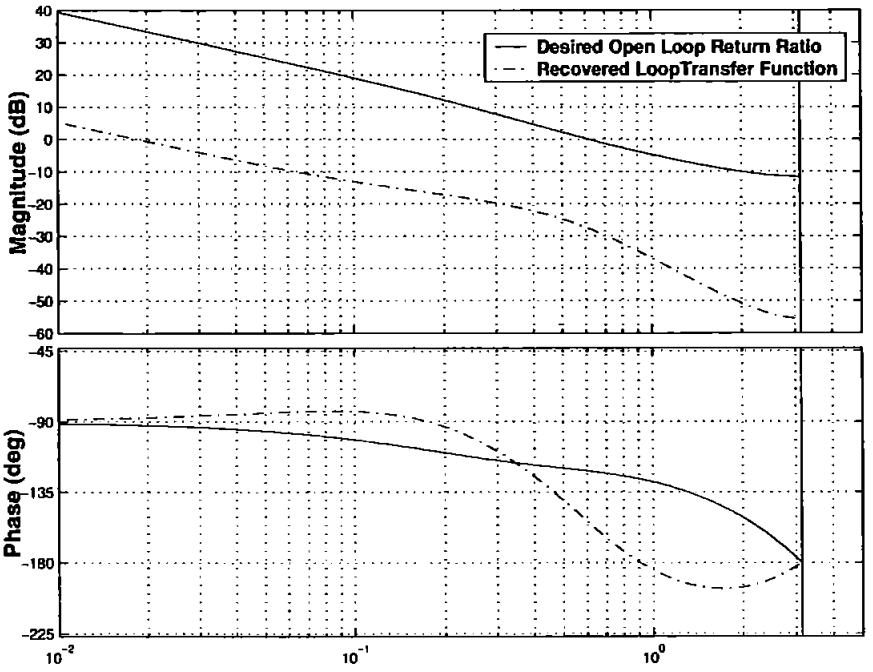


Figure 6.15: Stability characteristics of LQG/LTR controller showing deviation from the desired values due to increase in the magnitude of R

Waypoint following

Healey and Lienard (1993) proposed a guidance mechanism well known as waypoint following in the underwater research community. This has been discussed in detail in Chapter 2 and a short summary follows. In this method, the vehicle is required to follow several waypoints to arrive at the destination. A COA is defined around the waypoint so that when the AUV enters this region, the next waypoint is selected. This section presents the integration of LQG/LTR controller with the waypoint following scheme. As will be shown the waypoint following is essentially equivalent to multiple LOS trackings with the LOS angle calculated between any two given waypoints using Equation 5.1. There are four waypoints selected to perform the simulations which are provided in Table 6.1 below.

$x(m)$	100	300	500	500
$y(m)$	50	50	150	250

Table 6.1: Selected mission waypoints

In practice, the mission waypoint coordinates pre-stored in an onboard computer are compared with the coordinates of the vehicle obtained using a GPS provided that the vehicle is on the surface. For simulation purposes, no disturbances are assumed to be acting on the vehicle. The vehicle was launched at the point $(0, 10)$ and it quickly establishes a LOS angle between its launching position and first waypoint at $(100, 50)$. When the vehicle enters the COA, which in this case has a radius of $10m$, the second waypoint is selected. The LOS angle is now computed between the updated and last waypoints. Figure 6.16 depicts the vehicle trajectory and the mission waypoints. The AUV seems to be closely following the ideal track, however, the dynamics of the vehicle could be observed from the figure especially during turning manoeuvres. The rudder deflections required to achieve the waypoint following is shown in Figure 6.17. There is some rudder activity observed when the vehicle is manoeuvring for next way point. The standard deviation (STD) of the rudder movement is approximately 6^0 while the mean square error (MSE) between actual and ideal vehicle coordinates is $9.6m^2$. Finally the *Hammerhead* heading angle is illustrated in Figure 6.18 which provides evidence of the fact that the waypoint following is equivalent to multiple LOS tracking which are switched on the basis of COA.

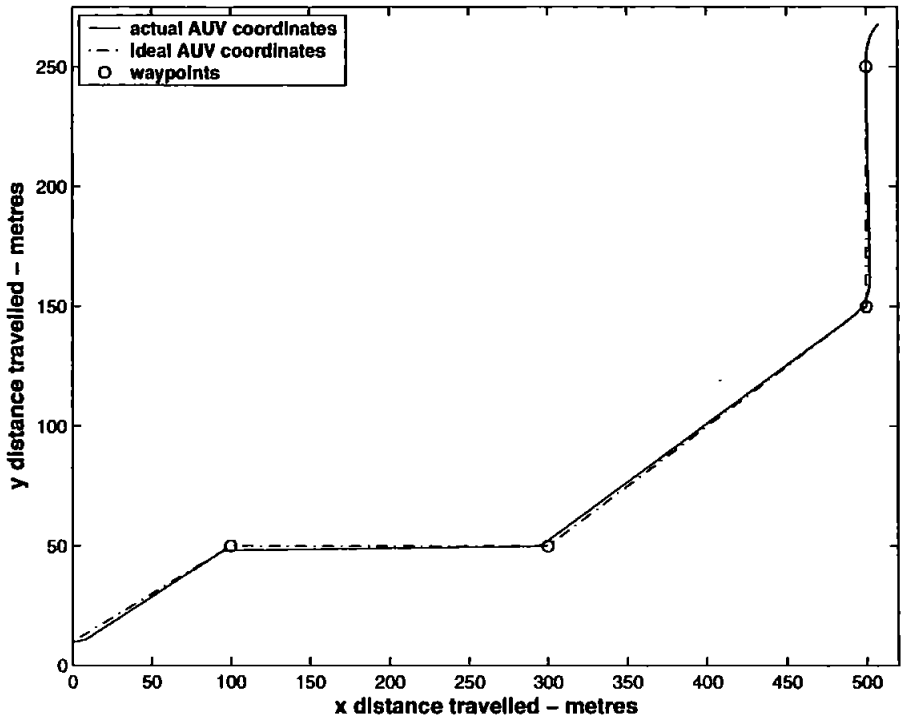


Figure 6.16: LQG/LTR control of *Hammerhead* heading showing AUV trajectory and waypoints

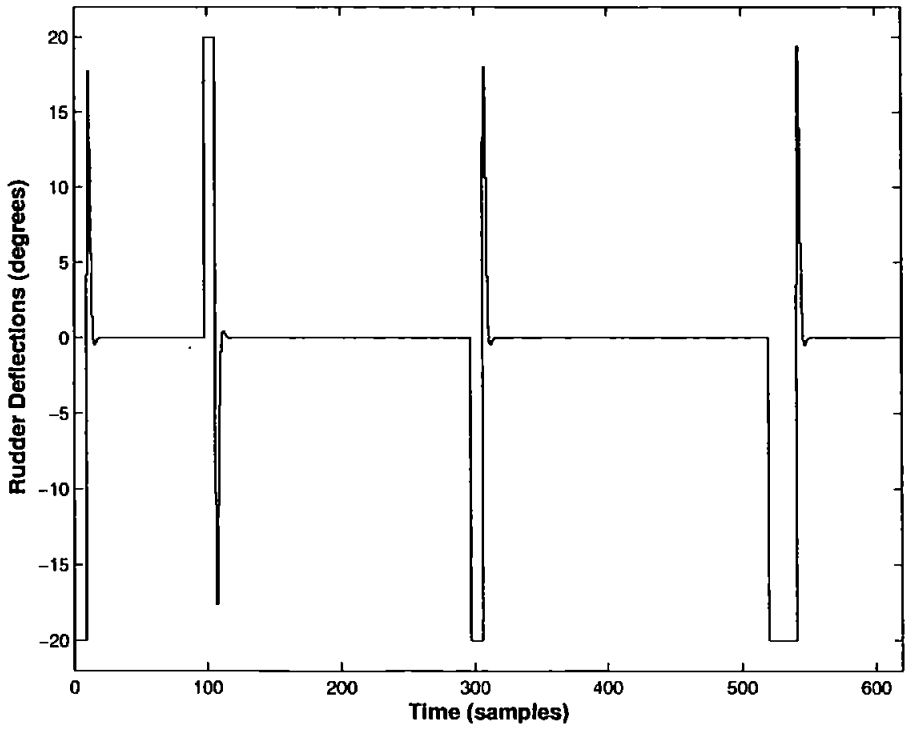


Figure 6.17: LQG/LTR control of *Hammerhead* heading showing control actions required to follow the selected waypoints

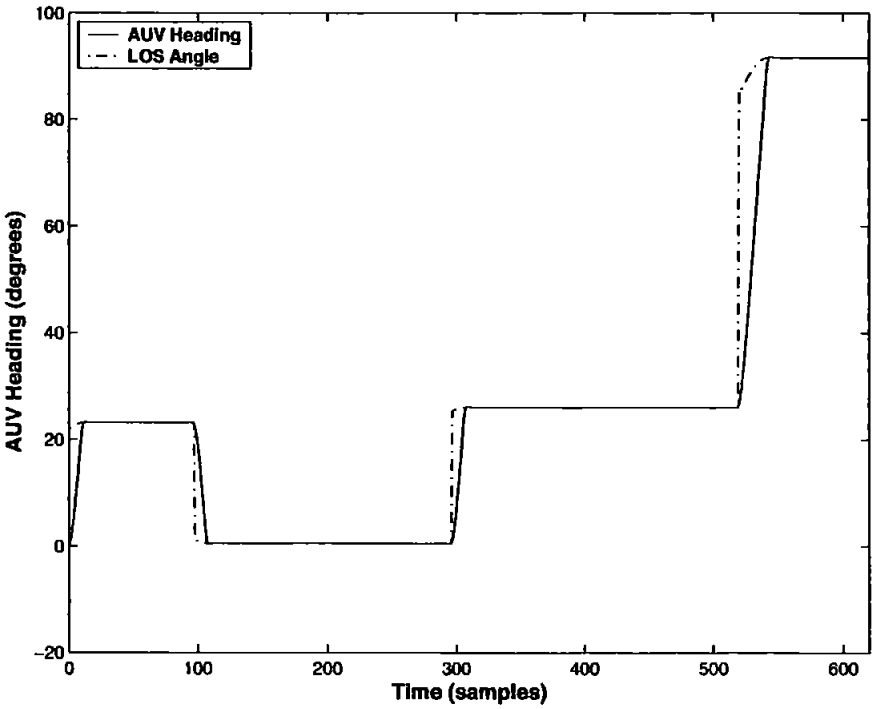


Figure 6.18: LQG/LTR control of *Hammerhead* heading showing the AUV is closely following the desired LOS angles

Next, the LQG/LTR based autopilot for *Hammerhead* AUV is simulated for waypoint following when there is a strong current acting on the vehicle in the positive y-direction. The current velocity is assumed to be $0.5m/s$ which is half the AUV velocity. The magnitude of the current velocity assumed is found to be adequate to determine the robustness of the controller since typical values encountered on a calm day are between $0.2 - 0.3m/s$ (Taylor, 2000). The vehicle's trajectory is depicted in Figure 6.19 showing the effects of current on the vehicle motion. Clearly, the disturbance is forcing the vehicle off the desired track, however, the AUV manages to reach within the COA of all the waypoints. The disturbance is observed to have a strong impact on the vehicle's movement especially when manouevring in the positive x-direction. A maximum deviation of approximately $80m$ is observed in this particular case. The MSE between the actual and ideal vehicle coordinates is approximately $698m^2$. The STD of the rudder movement is estimated at 9^0 which is depicted in Figure 6.20 showing that the autopilot is generating vigorous control actions to keep the vehicle on course. A heading plot of the AUV motion is illustrated in Figure 6.21 where the LOS angle is changing continuously due to the addition of currents. However, the vehicle follows the desired LOS closely and hence reaches all the target waypoints.

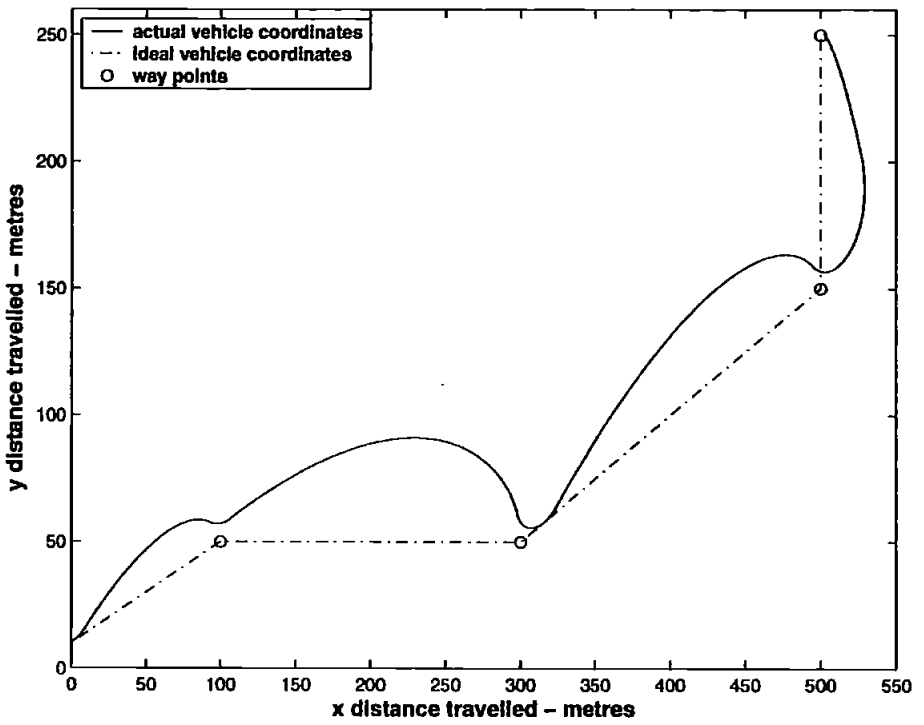


Figure 6.19: LQG/LTR control of *Hammerhead* heading showing AUV trajectory and waypoints with a sea current in the positive y-direction

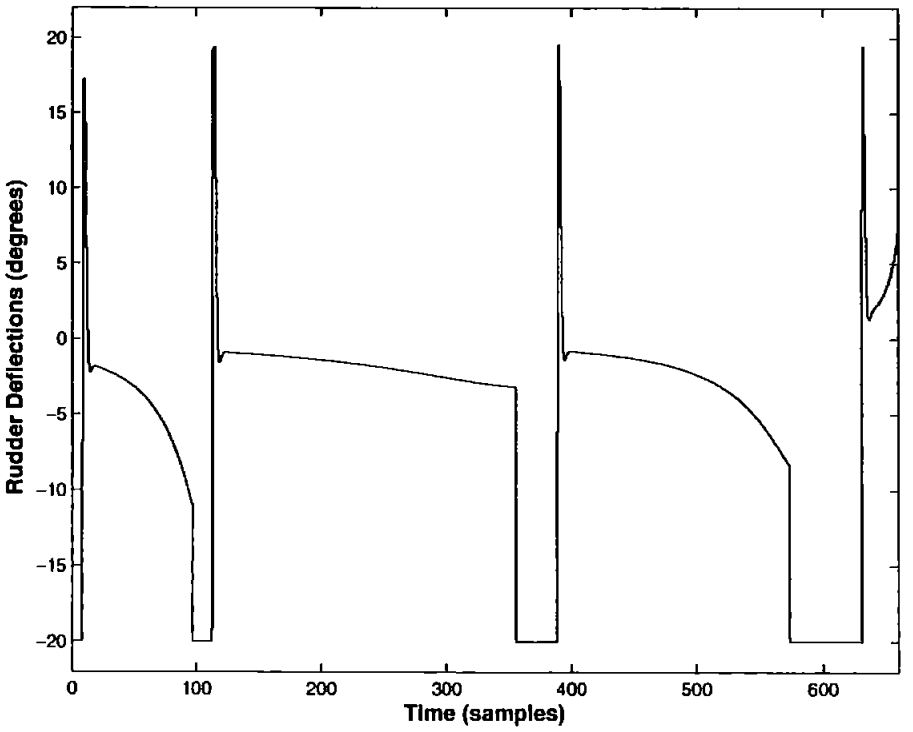


Figure 6.20: LQG/LTR control of *Hammerhead* heading showing control action required to follow the selected waypoints with a sea current in the positive y-direction

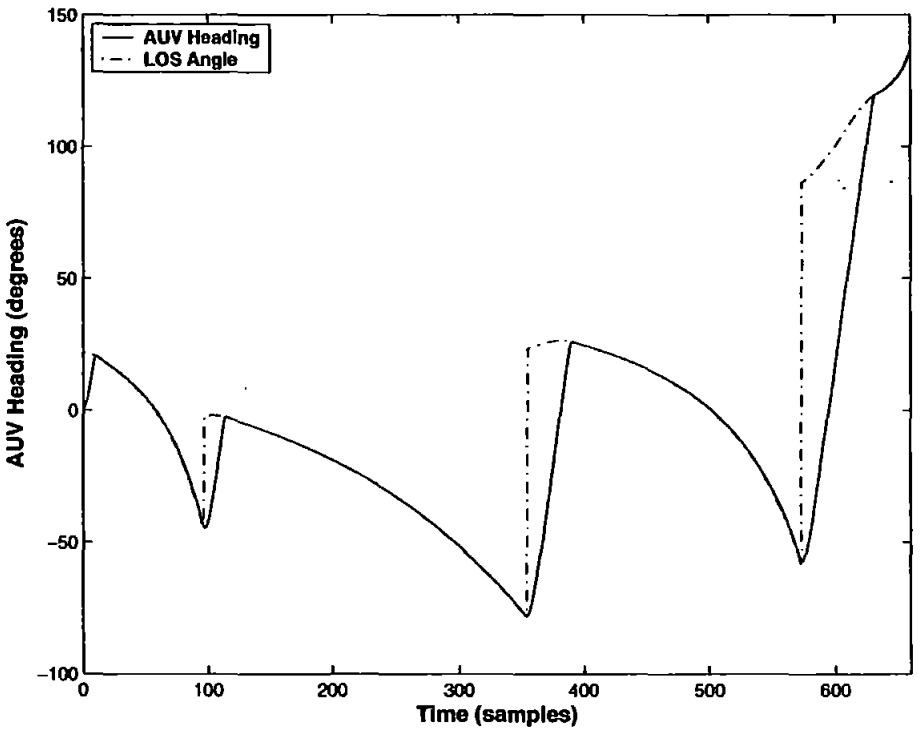


Figure 6.21: LQG/LTR control of *Hammerhead* for waypoint following showing the AUV is closely following the desired LOS angles with a sea current in the positive y-direction

Proportional navigation guidance

Chapter 5 proposed two guidance laws to provide commands for underwater tracking of a cable/pipeline. Simulation results were also shown in both cases which were obtained without considering the vehicle dynamics. In this section, the first of the two guidance strategies, namely, the PNG law is integrated with the LQG/LTR heading controller developed for the *Hammerhead* vehicle model given by Equation 4.19. The guidance law generate commands that are proportional to the LOS angle. The AUV and cable were assumed to be point masses moving with a constant velocity and an onboard active sonar picks the cable position even if it is buried. The AUV and cable position coordinates are then used to evaluate the LOS angle which is eventually utilised by the guidance law. The Simulink block diagram of the PNG system is shown in Figure 5.2. The input to the controller is the LOS angle represented as "theta1" and the variable "auv_heading" denotes the vehicle heading which is used along with the vehicle velocity to update guidance commands. The scenario considered here is similar to the one presented in Section 5.1.5 except that now the vehicle dynamics are

involved. The guidance system has been designed in Simulink whereas the controller is developed in Matlab which are then combined to form an integrated guidance and control system as shown in Figure 6.22.

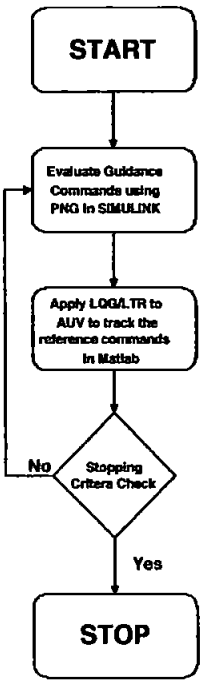


Figure 6.22: Flow chart of the integration of LQG/LTR and PNG systems developed in Matlab/Simulink environment

The AUV coordinates in the inertial two-dimensional frame of reference (x, y) plane are $(0, 200)$ with respect to the target FOC which is $(0, 10)$ representing the seabed. A bias is introduced after descending to a depth of $10m$ above the sea bed[2].

The result of the cable tracking mission is depicted in Figure 6.23 showing the effect of adding vehicle dynamics in the loop in comparison with Figure 5.3. The vehicle after launching from a mothership starts following the cable as soon as the onboard sonar receives the first echoed ping. By adding the $10m$ bias in the y-coordinate, the vehicle follows the cable at a fixed height without ever intercepting it thus providing an invaluable tool for cable/pipeline inspection purposes.

[2] $10m$ is quite high for real time applications but it was assumed here to validate the concept

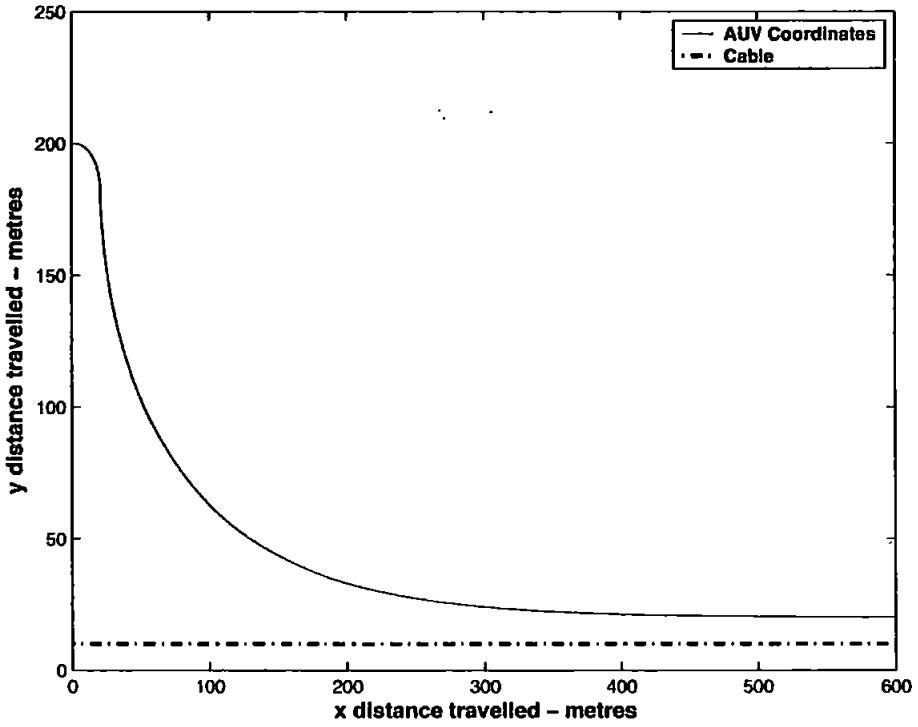


Figure 6.23: AUV and target position coordinates where the AUV is tracking the cable at a specified height

The PNG law updates the LOS angle at every sample time, and this is reflected in the plot shown in Figure 6.24 for the desired yaw and vehicle heading. Since the vehicle launching coordinates were $(0, 200)$ whilst the inspection site began at $(0, 10)$, therefore the initial desired heading angle when the vehicle was launched is $-90°$ as shown in Figure 6.24 which is subsequently updated as the vehicle approaches the cable. The optimal control surface deflections are also shown in Figure 6.25 and are within the specified actuator constraints. The positive actuator constraint is active during the first 10 samples times when the vehicle was making a turn followed by a spike from 20^0 to -20^0.

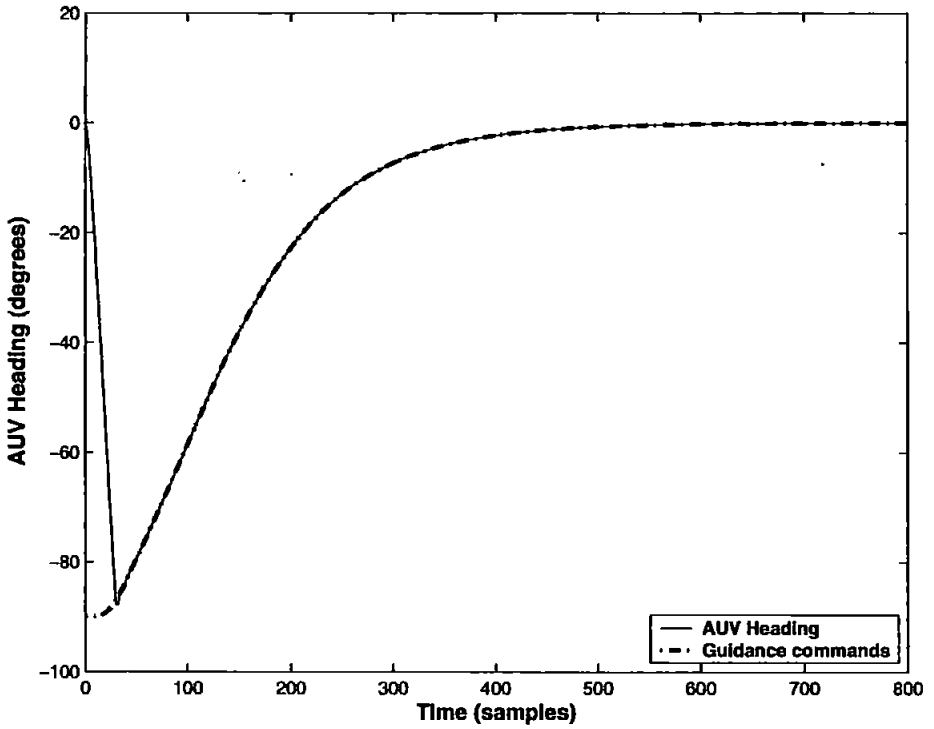


Figure 6.24: AUV heading controlled by the LQG/LTR autopilot and is shown to be closely following the guidance commands generated by the PNG

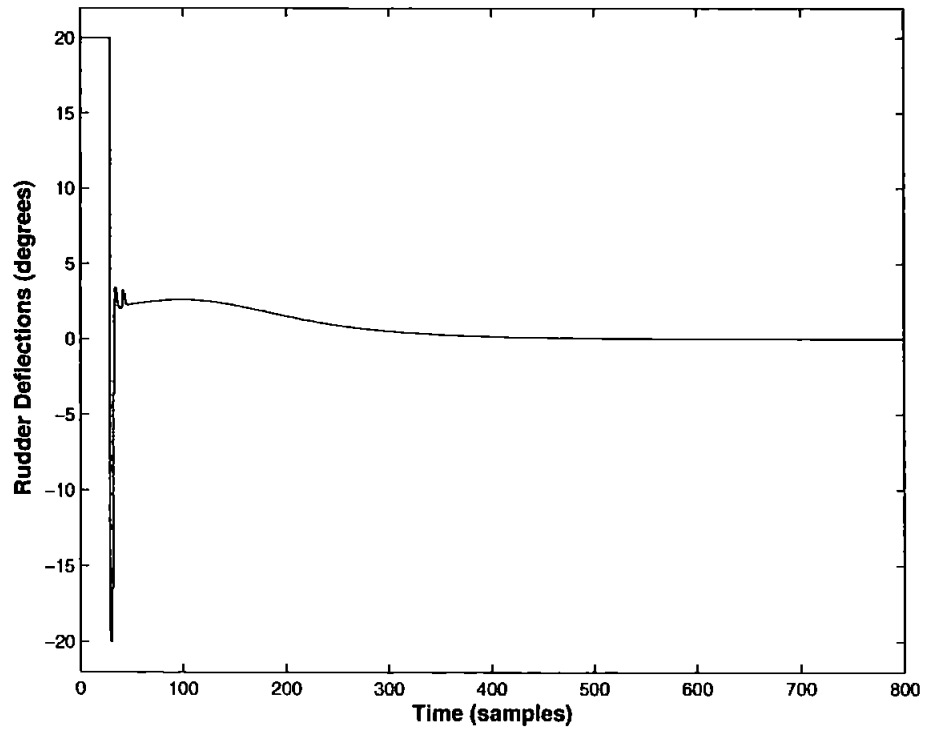


Figure 6.25: Rudder deflections evaluated by the LQG/LTR controller needed to track the reference trajectory generated by the PNG law

Integrated hybrid guidance and control system

A hybrid guidance law was proposed in Chapter 5, which was formulated on the basis of vehicle speed and distance from the target. The guidance law is specifically designed for cable tracking purposes and the trajectory generated to be followed by the AUV was shown in Figure 5.8. The following assumptions are taken for hybrid system simulation in addition to the two given at the beginning of Section 6.7

- The AUV is equipped with a vision system that generates the coordinates of the points on the cable to be tracked.
- The initial target coordinates (one end of the cable) are known prior to the mission.

The results presented herein have been published by Naeem *et al.* (2003*a*). The underlying idea of the hybrid guidance system is to reduce the vehicle velocity gradually as it approaches the target. This helps in reducing the overshoot and smoothes the AUV trajectory. An onboard vision system then takes over that provides the cable position coordinates in the final tracking phase of the mission. The complete problem formulation can be found in Section 5.2. It was mentioned that since the *Hammerhead* model is not available at different speeds, therefore an AUV model has been borrowed from the literature for simulation purposes whose parameters can be obtained at various velocities for a proof of concept study. The AUV test model is that used by Kwiesielewicz *et al.* (2001), where the model parameters are given in terms of vehicle speed. The SISO vehicle model for a given vehicle velocity can be described by the following transfer function,

$$G(s) = \frac{as + b}{s(s^2 + cs + d)} \tag{6.33}$$

where the coefficients a, b, c and d are defined based on vehicle speed v in knots.

$$a = 0.05803v^2, \qquad b = 0.00449v^3$$
$$c = 0.25963v, \qquad d = 0.00856v^2$$

The input to the AUV are the rudder deflections whereas the output is the heading of the vehicle. The model parameters are calculated at three different vehicle speeds i.e. 5, 7.5 and 10 knots and the resulting continuous time model is discretised at a sampling rate of $10Hz$. The AUV model is assumed to have a turning radius of $25m$ and the constraints on the rudder actuator are 25^0 in either left or right direction. The

AUV model contains a pole on the unit circle i.e., type 1 system, hence the contents of the feedforward block $\mathbf{N}_u$ will be zero. Separate controllers are developed one for each vehicle speed model and switched on the basis of vehicle distance from the waypoint which in this case is the known end of the cable. Herein, the COA is assumed to be at least the minimum turning radius of the vehicle while the circle around the waypoint has a radius chosen to be at least twice the length of the vehicle. The vehicle speed is reduced as it crosses each circle boundary from midcourse to terminal and from terminal to tracking phase.

The desired controller specifications for this AUV model are chosen as 12 dB for the GM and 58^0 for the PM. The controller parameters are tuned separately for all the models to obtain the same desired response for each autopilot and are provided in Table 6.2. Only $\mathbf{R}_w$ is varied to achieve the desired frequency domain characteristics whereas optimal values are chosen for the remaining parameters.

Controller	Vehicle speed		
Parameters	5 knots	7.5 knots	10 knots
$\mathbf{R}_w$	$169\mathbf{I}_2$	$39.2\mathbf{I}_2$	$14.5\mathbf{I}_2$
$\mathbf{R}_v$	1	1	1
$\mathbf{Q}$	$\mathbf{C}^T\mathbf{C}$	$\mathbf{C}^T\mathbf{C}$	$\mathbf{C}^T\mathbf{C}$
R	≈ 0	≈ 0	≈ 0

Table 6.2: LQG/LTR controller parameters for various speed models

The Bode plot of the desired filter's open-loop return ratio for the 10 knots speed model of the vehicle is shown in Figure 6.26. The GM, PM and gcf can be readily evaluated from the plot. The next step is to calculate the feedback gains using the optimal $\mathbf{Q}$ and R in Equations 6.28 and 6.29 respectively and finally develop the LQG/LTR compensator using Equation 6.26. The loop transfer function $\mathbf{G}(z)\mathbf{H}(z)$ is also evaluated and the Bode plot superimposed on the Bode plot of the open-loop return ratio in Figure 6.27 shows the amount of recovery achieved. In this case, full recovery is achieved as the two plots coincide with each other. Figure 6.28(a) presents the step response of the closed loop feedback system showing a large overshoot. This can be reduced by adding more damping to the system by introducing a weighting factor on the diagonal term of $\mathbf{Q}$ corresponding to the velocity state. This is equivalent to using rate feedback for improving damping from a conventional sense (Weerasooriya

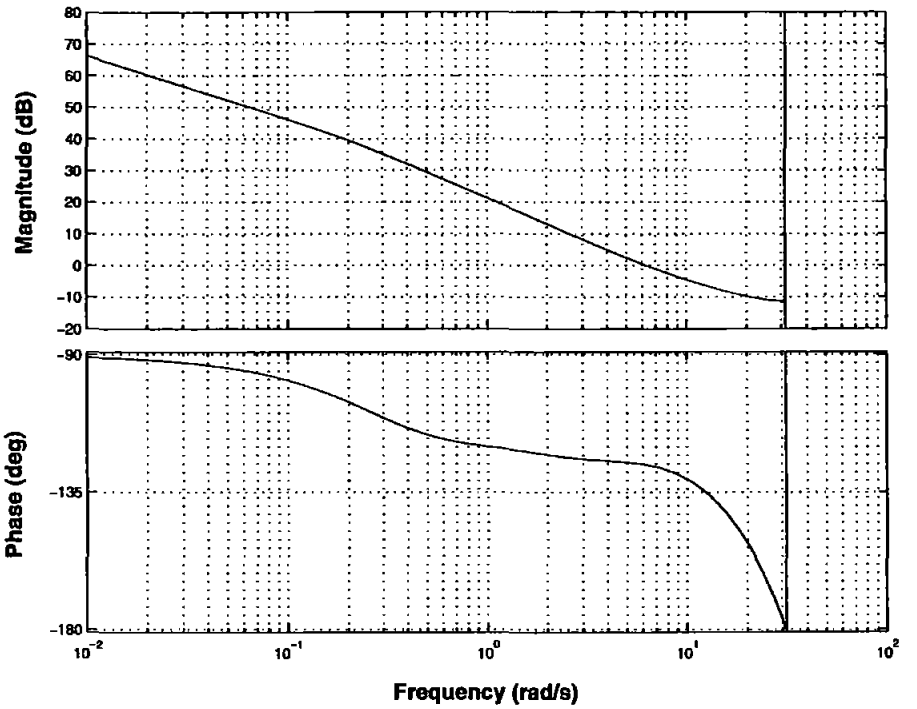


Figure 6.26: Bode plot of the target's filter open-loop return ratio

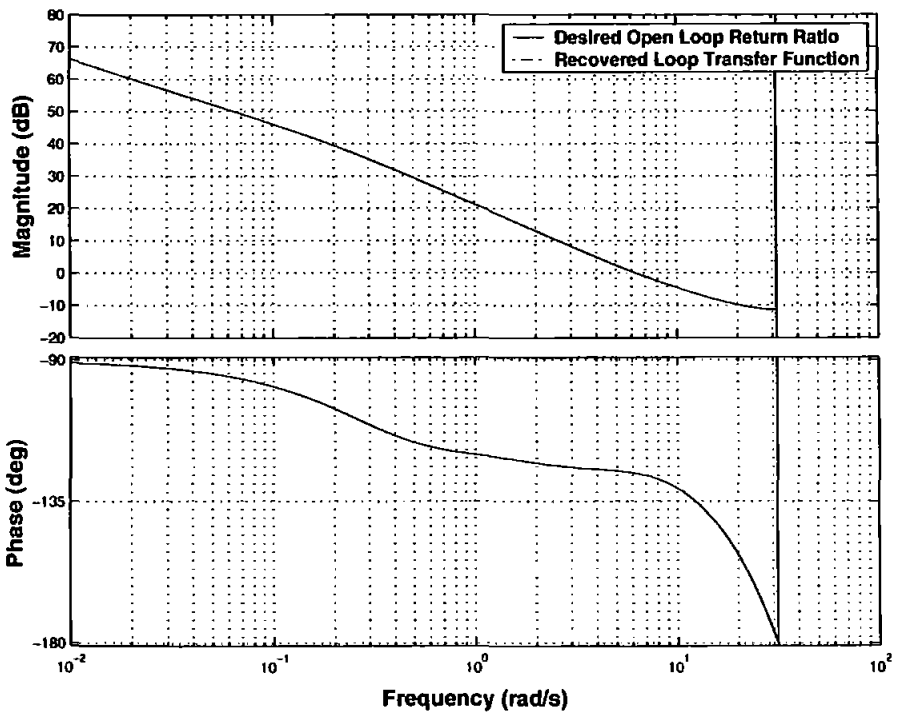
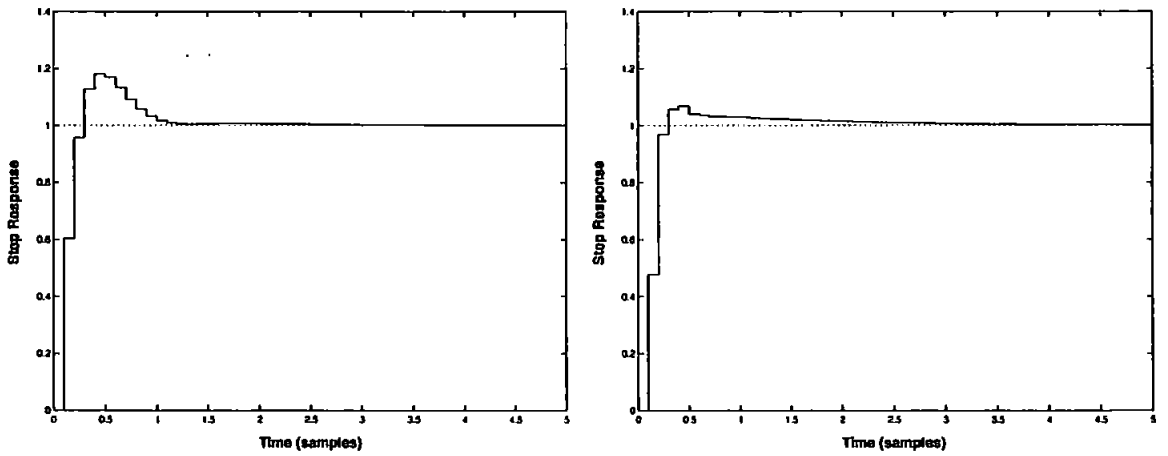


Figure 6.27: Bode plots of the filter's open-loop return ratio and recovered loop transfer function for nominal Q (full recovery)

and Phan, 1995). Figure 6.28(b) depicts the step response of the closed loop system with modified $\mathbf{Q}$ and Figure 6.29 presents the Bode plot of the loop transfer function.



(a) Step response of the closed loop system for $\mathbf{Q} = \mathbf{C}^T\mathbf{C}$ and $R \approx 0$

(b) Step response of the closed loop system for modified $\mathbf{Q}$ and $R \approx 0$

Figure 6.28: Step response of the closed loop depth autopilot for different values of the input weighting matrix R

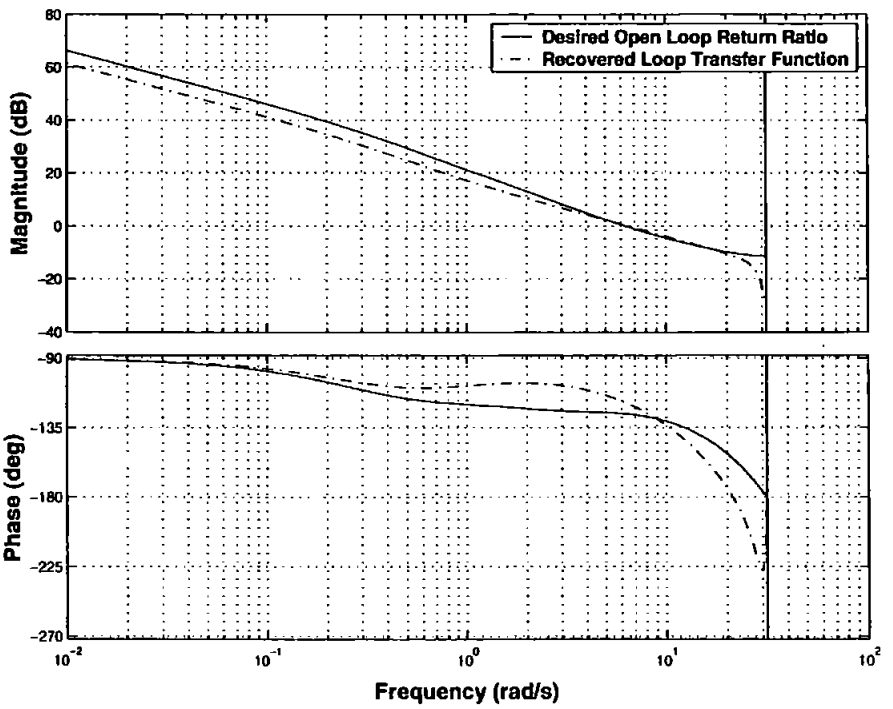


Figure 6.29: Bode plots of the filter's open-loop return ratio (solid line) and recovered loop transfer function (dashed line) with added damping (reduced stability margins)

Although overshoot has subsided it has been achieved at the cost of reduced stability margins due to the deviation from the optimal values. The same procedure has been adopted for all vehicle models at various speeds and compensators are developed. Finally guidance and control system integration is performed and the simulation results are shown in Figure 6.30 for a cable tracking mission, which clearly shows good tracking behaviour using the proposed guidance algorithm. The figure compares the vehicle trajectory when the speed is constant to the vehicle's path when the speed is varied based on the range from the cable. The result undoubtedly shows that the vehicle's trajectory becomes quite smooth using the hybrid guidance system and settles down quickly. Conversely, keeping the velocity at a fixed value is disadvantageous as there is plenty of manoeuvring involved before the vehicle started to track the cable which represents a waste of control energy.

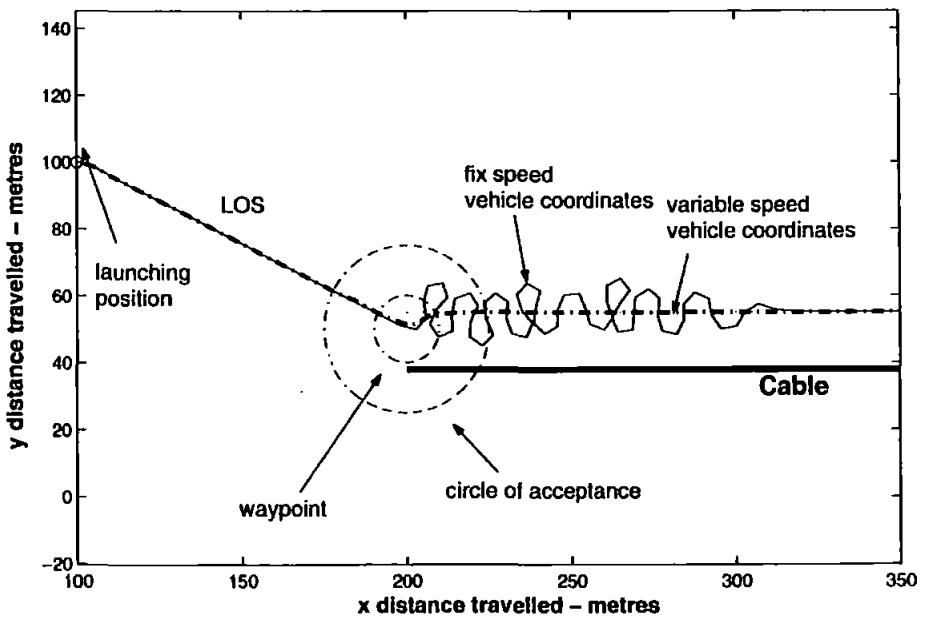


Figure 6.30: Cable tracking mission from launching to tracking, variable speed vs. fixed speed

The control surface deflections generated by the autopilot is depicted in Figures 6.31 and 6.32 for the case of optimal and modified **Q** respectively. Clearly, the modified **Q** with additional damping causes less variation in the control input as compared to optimal **Q** but at the cost of reduced stability margins. However, both figures suggest that the deflections are within the constrained actuator limits imposed by the saturation block.

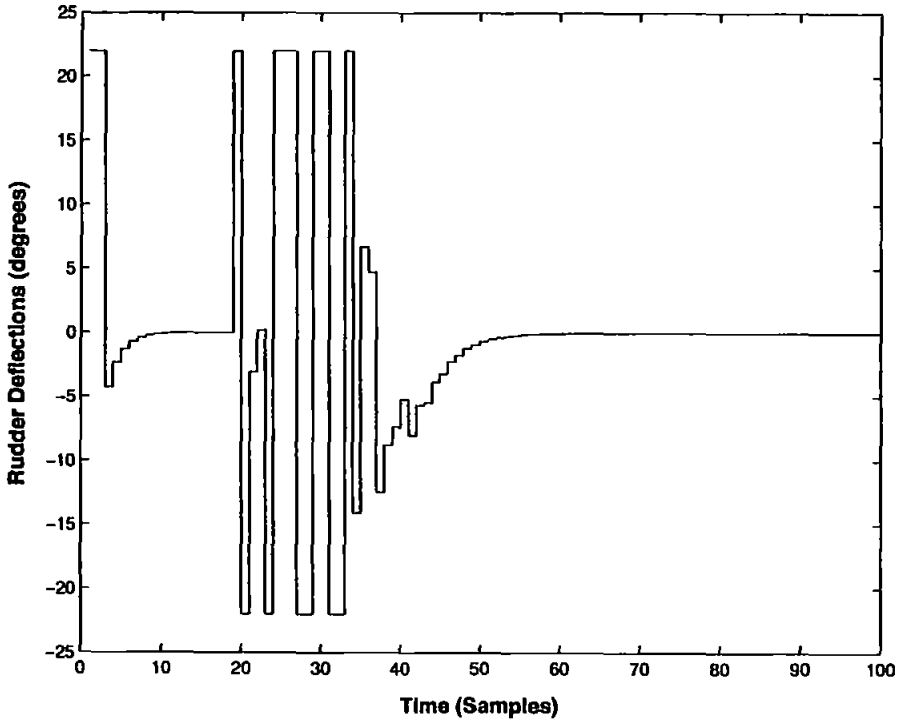


Figure 6.31: Rudder deflections generated by the LQG/LTR controller for $\mathbf{Q} = \mathbf{C}^T\mathbf{C}$ and $R \approx 0$

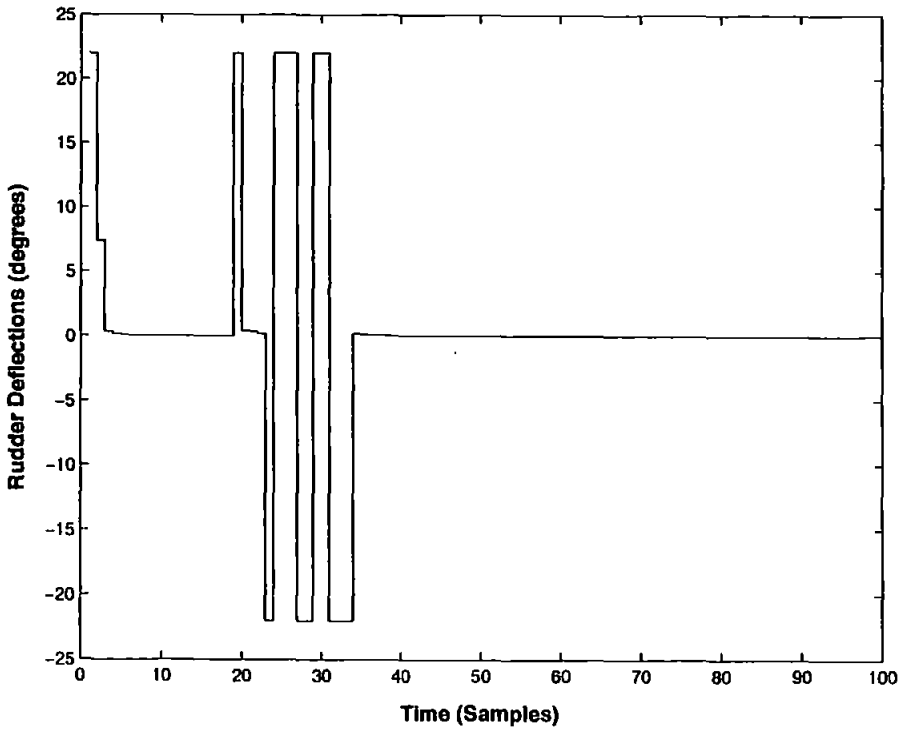


Figure 6.32: Rudder deflections generated by the LQG/LTR controller for modified $\mathbf{Q}$ and $R \approx 0$

6.7.2 Depth control

The preceding section explored a great deal of *Hammerhead* yaw dynamics. Several simulation results are presented to gauge the model or in other words, the *Hammerhead* response in the horizontal plane. The LQG/LTR controller was rigorously tested in simulations and was shown to be quite robust in the presence of disturbances and the performance was found to be adequate for a diverse range of situations.

Consideration is now been given to the development of a depth autopilot for the *Hammerhead* AUV. The depth dynamic model extracted in Chapter 4 using SI is used to tune the controller parameters. Again, the state space version of the model is employed owing to the requirement of the state feedback controller and Kalman filter. The model has a pole on the unit circle and exhibits a ramp output in response to a step input. The *Hammerhead* AUV speed is approximately 2 knots and the overall nature of the response is rather slow. Therefore large settling times are likely to be observed in the depth plots. Moreover, the vehicle is slightly positively buoyant which could be overcome by applying large amount of force in order to submerge. By the same token, resurfacing or upward manouevring should require less force for zero pitch angle. The force applied is through the front canards so a larger force means large deflection angle and vice versa. The maximum deflection angle of the hydroplanes is $\pm 25^0$ where positive angle is for diving and negative for resurfacing.

After the controller design is complete, its performance will be assessed in simulations for two simple cases. The first is a diving manoeuvre where the vehicle is initially assumed to be on the surface and is required to reach and maintain a certain depth specified as a mission parameter. Multiple depth commands will then be issued one after another when the vehicle is already at a certain depth and is required to move up and down to some specified depth levels. It is shown that the same depth autopilot could also be used to maintain a certain altitude which is validated by simulation results when a vision based guidance system provides the necessary altitude information.

Depth controller specifications

The methodology adopted for the depth controller design is similar to the way the heading autopilot was developed. All that is needed to specify is the GM and PM of the system. The process noise covariance matrix $\mathbf{R}_w$ is then tuned to achieve the de-

sired response assuming nominal values for other parameters as discussed previously. The desired open loop frequency domain specifications are given as 18 dB and 60^0 for the GM and PM respectively.

To attain the specified response characteristics, $\mathbf{R}_w$ is adjusted to $75\mathbf{I}_4$, where $\mathbf{I}_4$ is a 4×4 identity matrix. The Bode plot of the open loop return ratio $\Phi(z)$ is shown in Figure 6.33 which depicts these quantities. The LQR weighting matrices $\mathbf{Q}$ and R are then chosen according to Equations 6.28 and 6.29 and the frequency characteristics of the loop transfer function $\mathbf{G}(z)\mathbf{H}(z)$ is also drawn on the same plot in Figure 6.33. It is clear that full recovery is achieved as the two plots overlap each other. The feedback controller is finally formed and the unit step response is illustrated in Figure 6.34.

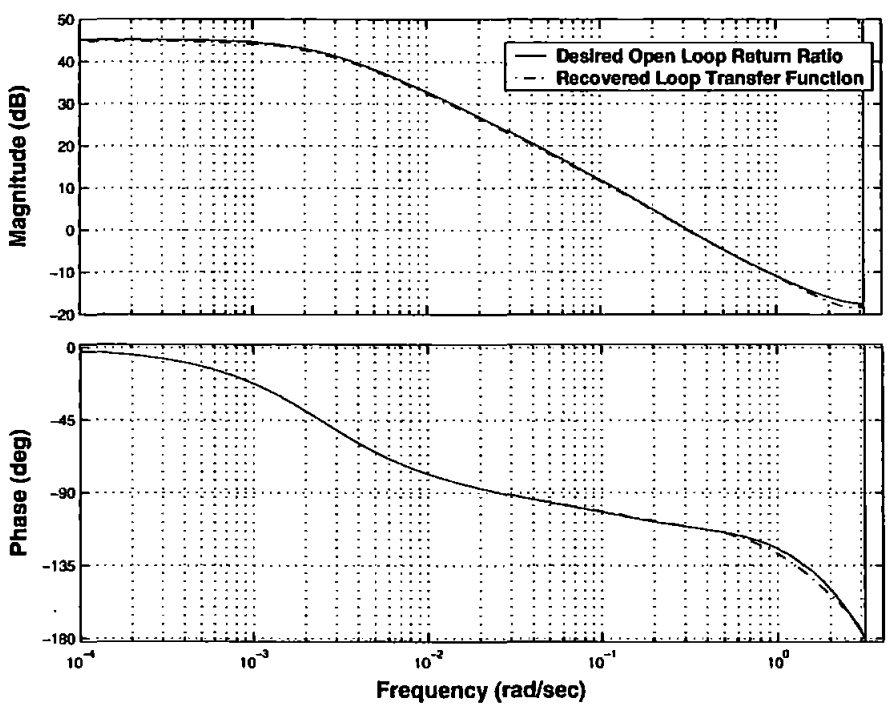


Figure 6.33: Bode plot of the open loop return ratio and loop transfer function of the depth autopilot showing amount of recovery achieved

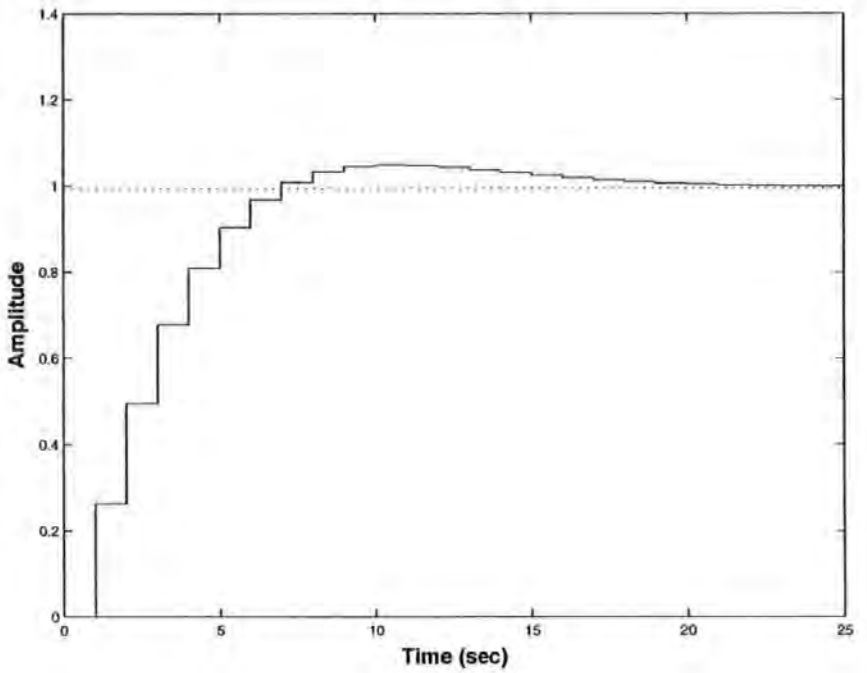


Figure 6.34: Unit step response of the LQG/LTR depth controller

There is a small overshoot of less than 5% which was deemed adequate for controller testing and the settling time is less than 20 seconds. The Simulink block diagram of the LQG/LTR controller is depicted in Figure 6.35, where "ref_depth" is a mission parameter supplied by the operator or the guidance system.

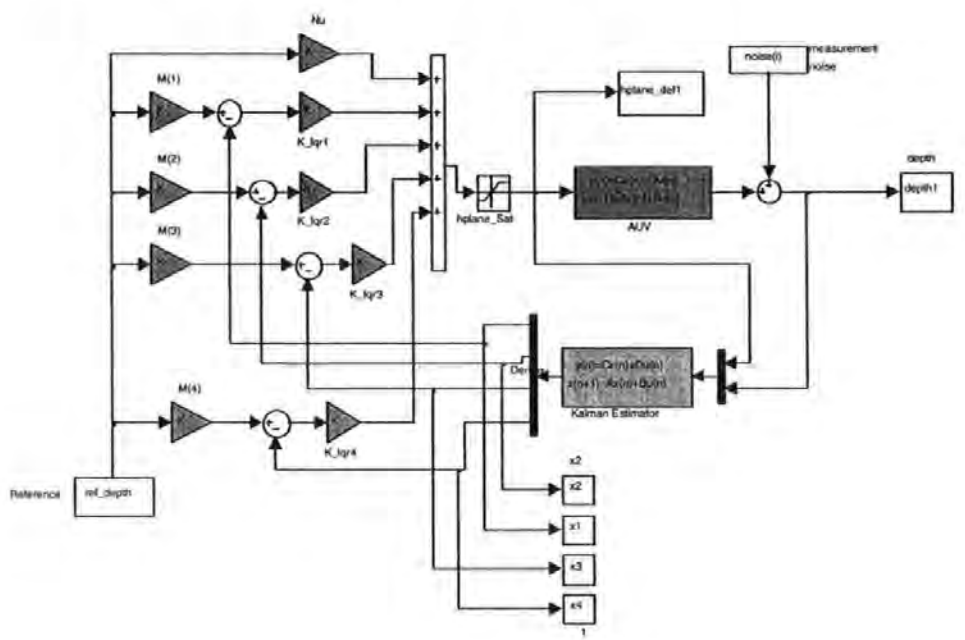


Figure 6.35: Simulink block diagram for the LQG/LTR based depth autopilot

The vehicle is launched on the surface and the task is to follow a depth of $3m$ below the sea surface. The simulation is run for 100 sample times and the *Hammerhead* response is depicted in Figure 6.36. With a small overshoot of less than 5% and settling time approximately 15 samples, the vehicle successfully follows the desired depth and stays on course throughout the rest of the mission duration. The control effort is also shown in Figure 6.37 which attains its steady state value of 0^0 in less than 20 samples. The positive constraint is active in the beginning for approximately 5 samples when the vehicle was diving and when the vehicle recovers from the overshoot. This could easily be avoided by increasing the magnitude of the control weighting matrix but at the cost of reduced stability margins.

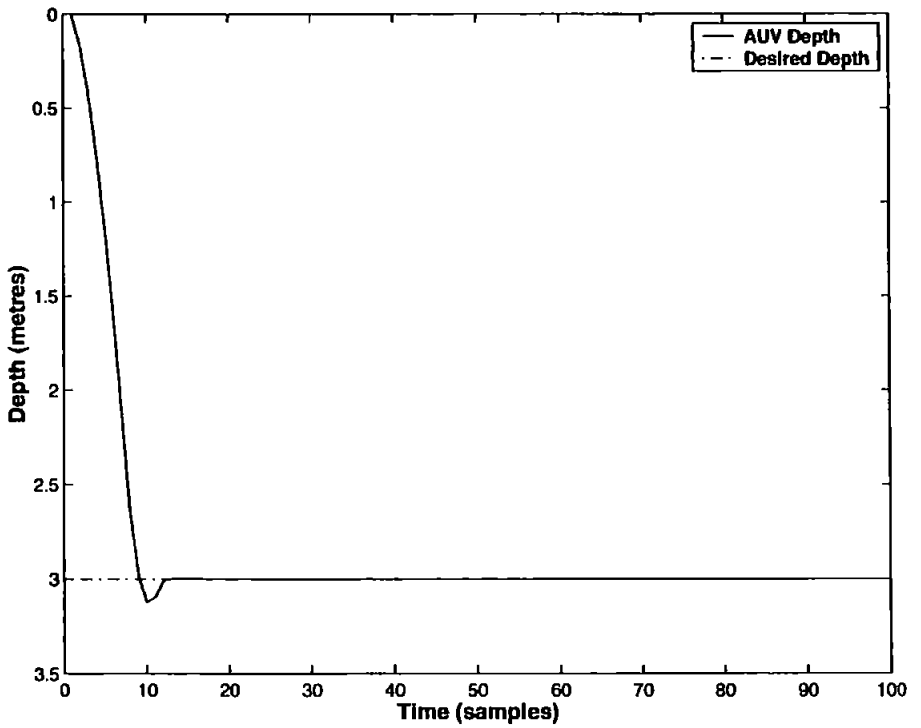


Figure 6.36: Depth control of *Hammerhead* vehicle using the LQG/LTR controller showing a step change in depth

The controller performance is now gauged for multiple depth command tracking. As before, the vehicle is launched on the surface and is required to undergo several changes in its depth. The "ref_depth" vector is now $[3\ 5\ 4\ 2\ 5]m$ which means that the vehicle will have to perform diving and surfacing manoeuvres. Each of the desired depth is kept for 50 sample times followed by another step change. The output response is depicted in Figure 6.38 which shows excellent controller performance to

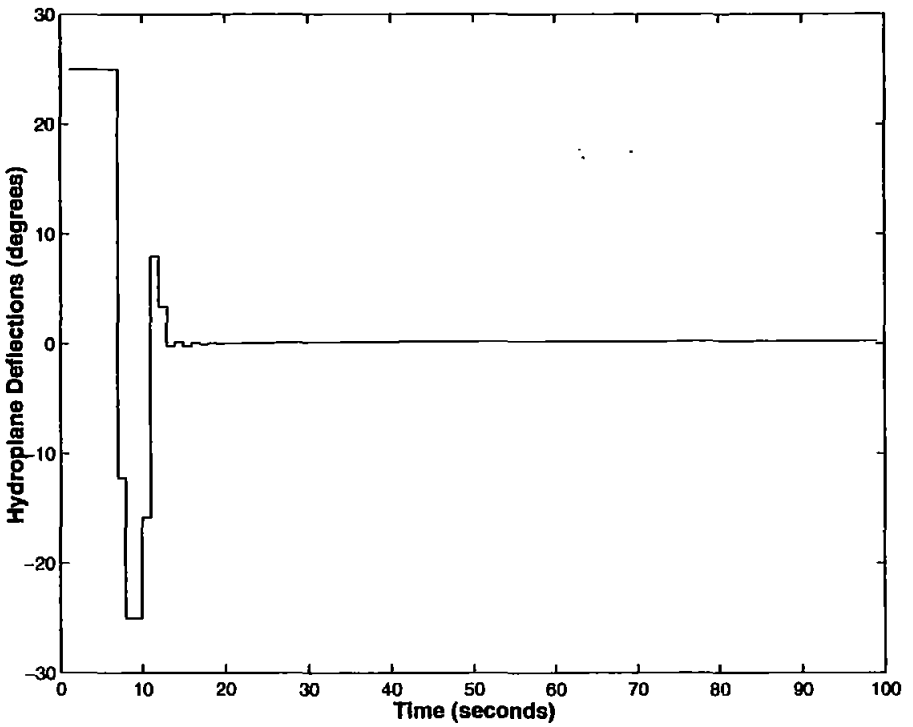


Figure 6.37: Hydroplane deflections required to achieve the specified depth

track multiple set point changes in depth. The small overshoot is evident from the figure which has been discussed before. The controller output response or the hydroplane deflections are plotted in Figure 6.39 showing that the vehicle is undergoing changes in its depth. The constraints are active when the AUV manoeuvres for each depth change. However, as mentioned previously, the vehicle requires relatively small amount of force for resurfacing as compared to diving. This can be observed from the deflection plot at approximately 50 and 150 sample times when the vehicle depth change is $2m$ from 3 to $5m$ and from 4 to $2m$ respectively.

Altitude control

The altitude control is a corollary to the depth control and is therefore included in the same section. The *Hammerhead* AUV currently has an onboard pressure sensor to measure the depth of the vehicle below the sea surface. A laser stripe illumination based vision system is also installed on the vehicle which has been developed at Cranfield University. In addition to the other parameters such as velocity and images, the vision system provides the altitude of the vehicle above the sea bed. An altitude sonar is also procured whose output is used to assess the vision system performance.

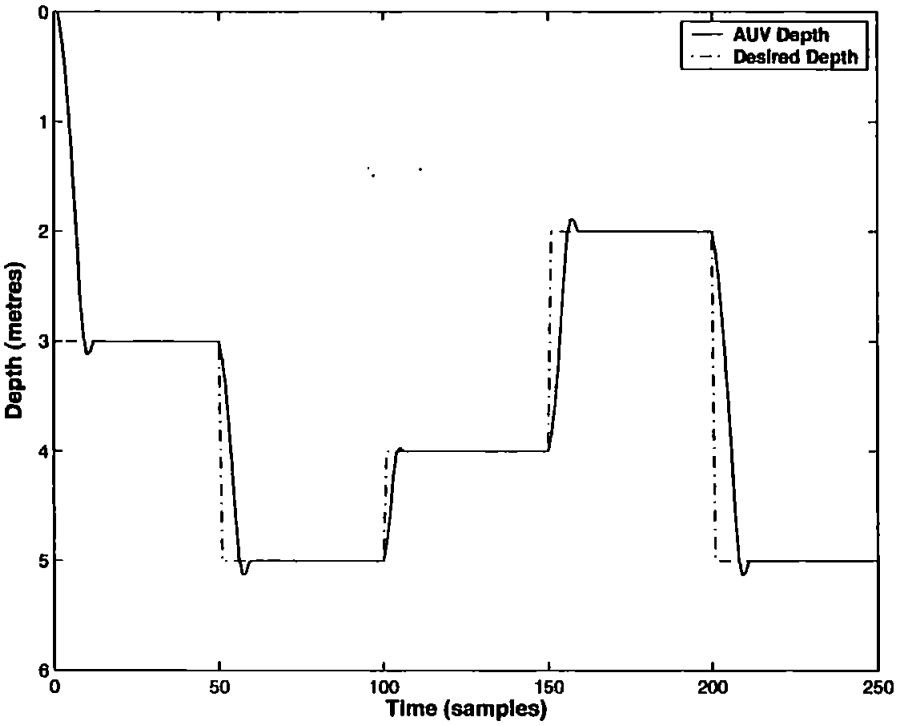


Figure 6.38: LQG/LTR depth control of *Hammerhead* vehicle showing multiple changes in depth command

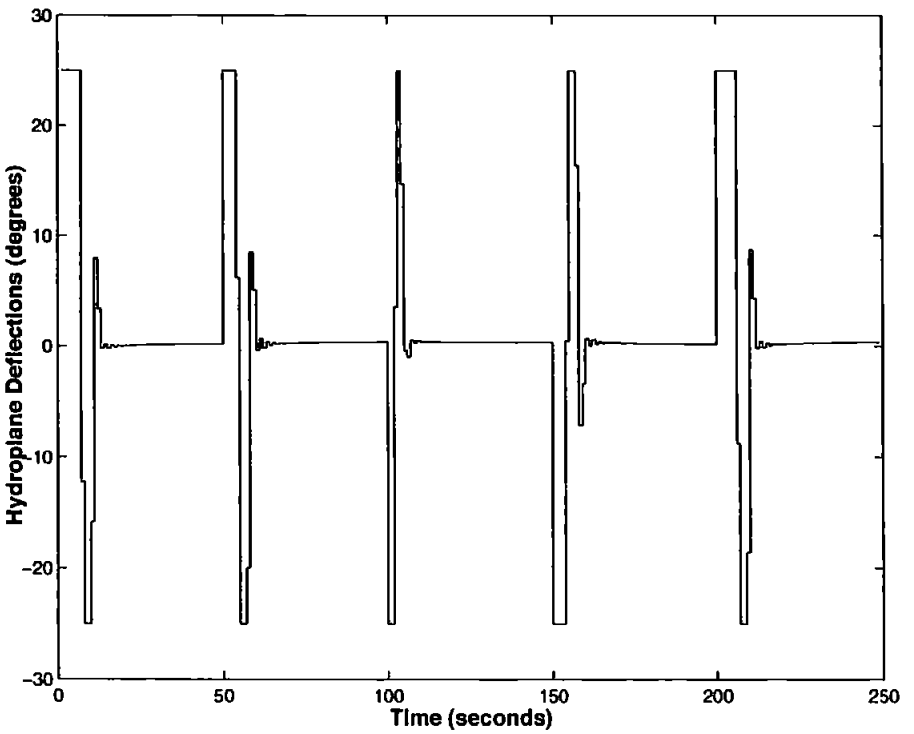


Figure 6.39: LQG/LTR depth control of the *Hammerhead* AUV showing control surface deflections needed to maintain the specified depth demands

There are several cases when there is a need to follow a certain altitude rather than depth. For instance, to maintain a certain image quality under water, the vision system may ask the control system to manoeuvre the vehicle in the vertical plane.

In this section, it is shown that the same depth autopilot developed in the preceding subsection can be used as an altitude controller. For this purpose, the depth and altitude of the vehicle obtained from the pressure sensor and vision system respectively are added to get an estimate of the depth of the sea bed. The demanded altitude is then subtracted from the sea depth to evaluate the desired depth which acts as an input to the feedback controller. It is also assumed that the vision guidance system is not perfect and there are errors in the altitude information which translates into uncertain depth demands. For simulation purposes, it is presumed that the sea depth and altitude information are available which on subtraction provides the required depth. A normally distributed random noise of zero mean and variance 0.001 is added to the depth (altitude) commands to simulate the uncertainty in the vision system output. The depth controller is simulated with the same parameters and the result is shown in Figure 6.40.

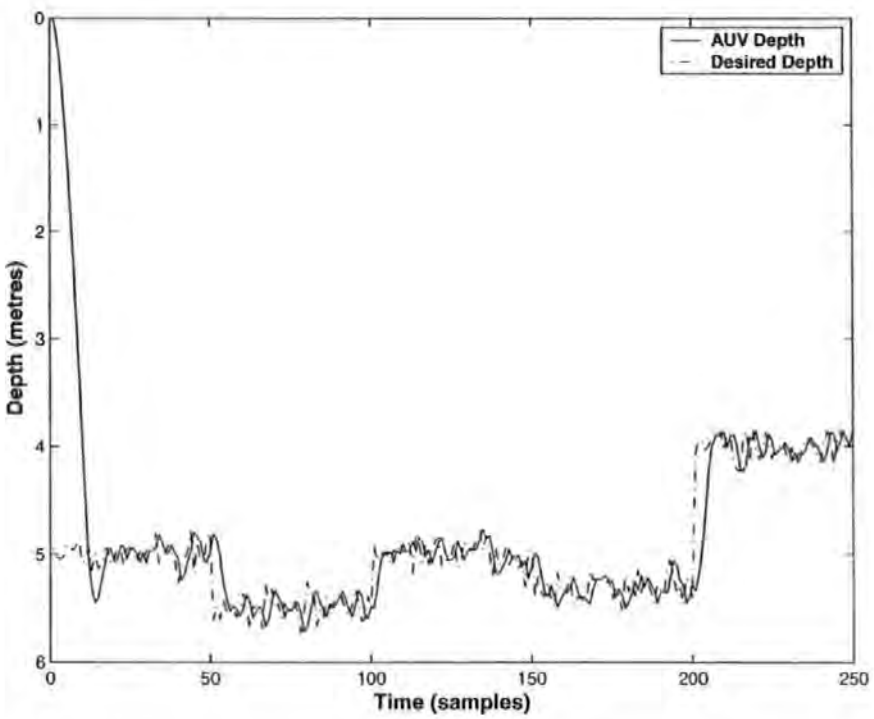


Figure 6.40: LQG/LTR depth control of *Hammerhead* using the altitude information when integrated with an onboard vision system

Due to the noise in the guidance input, the depth demand is changing at every sampling instant, however, the vehicle is closely following all variations in the guidance input. In practice, this is not desirable as the control energy is being wasted as depicted in Figure 6.41 showing vigorous control effort to maintain the specified altitude (depth). The standard deviation of 12.4^0 clearly tells the story which desperately needs to be reduced. One way is to add a low pass filter in series with the vision system output. Another simple remedy is to increase the magnitude of R which is adopted here. However, the increase in R lowers the stability margins shown in Figure 6.42 and a low pass filter would probably be a better choice in this case.

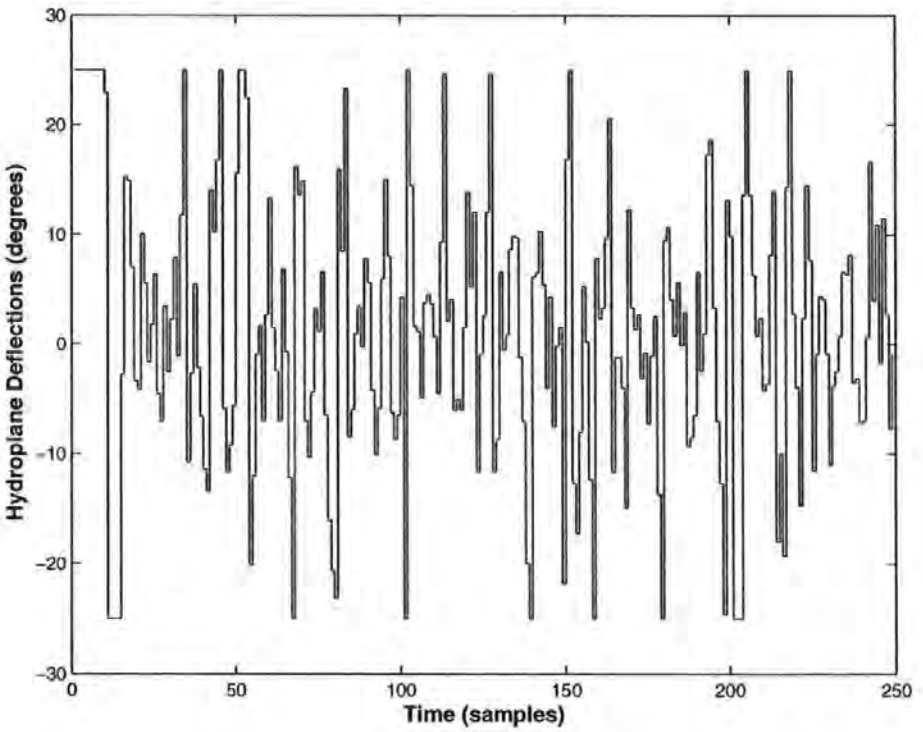


Figure 6.41: Hydroplane deflections needed to follow the desired depth when the vision system is integrated with the depth autopilot

Finally, the depth response of *Hammerhead* vehicle when the magnitude of R has increased is shown in Figure 6.43. The response is much smoother as compared to the previous case. The hydroplane deflections plotted in Figure 6.44 are quite less vigorous and has a standard deviation of less than 6^0 which is nearly half of the previous case. There is one peak control surface movement at approximately 200 sample times when the vehicle is changing its depth from 5.5 to $4m$.

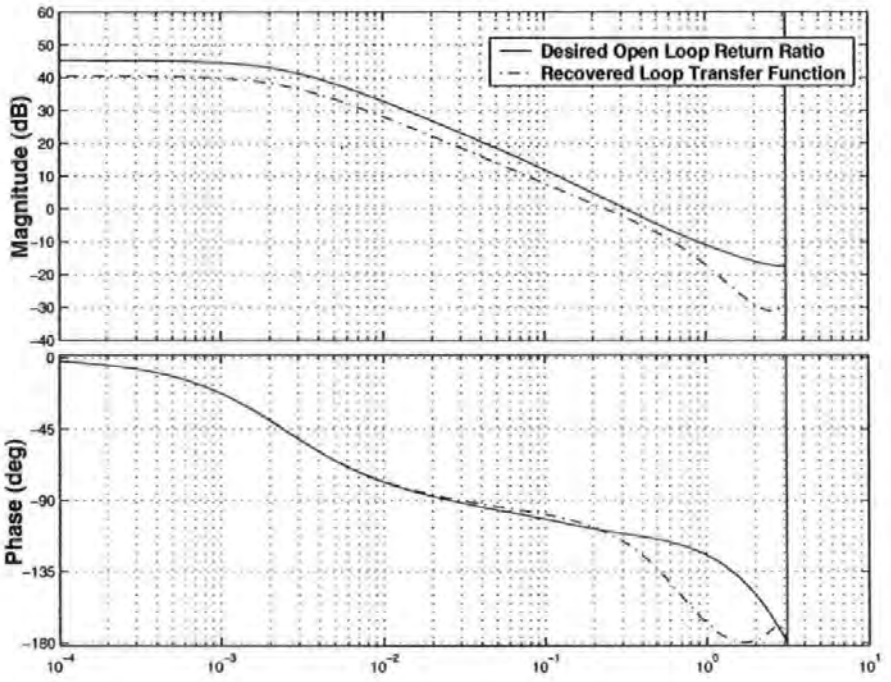


Figure 6.42: Bode plot showing the effect of changing R on depth controller performance

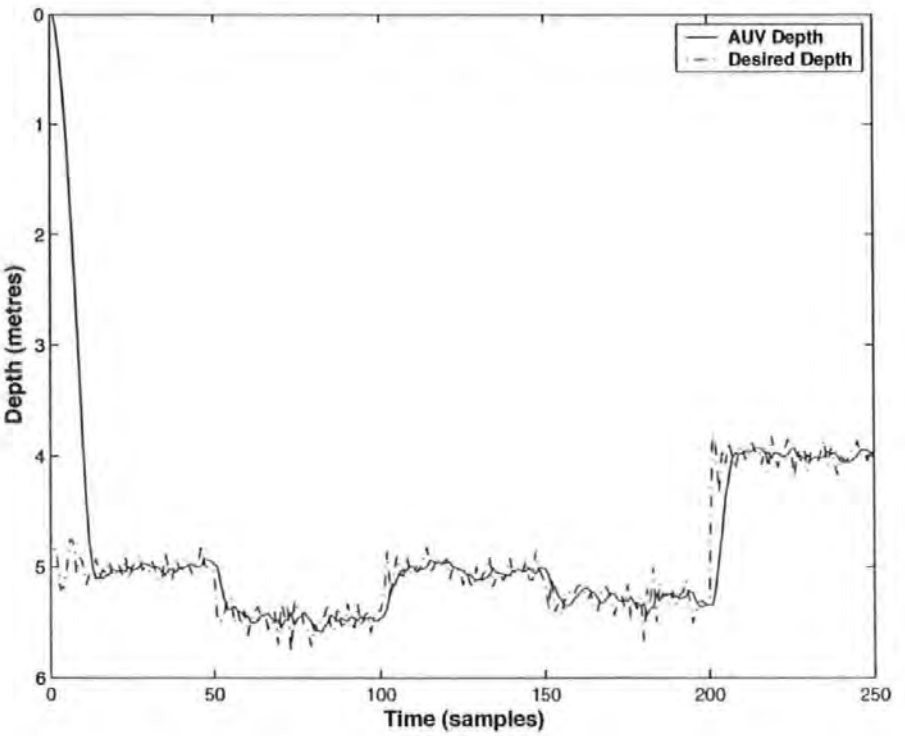


Figure 6.43: Depth control of *Hammerhead* AUV when the vision system is integrated with the autopilot and magnitude of R is increased

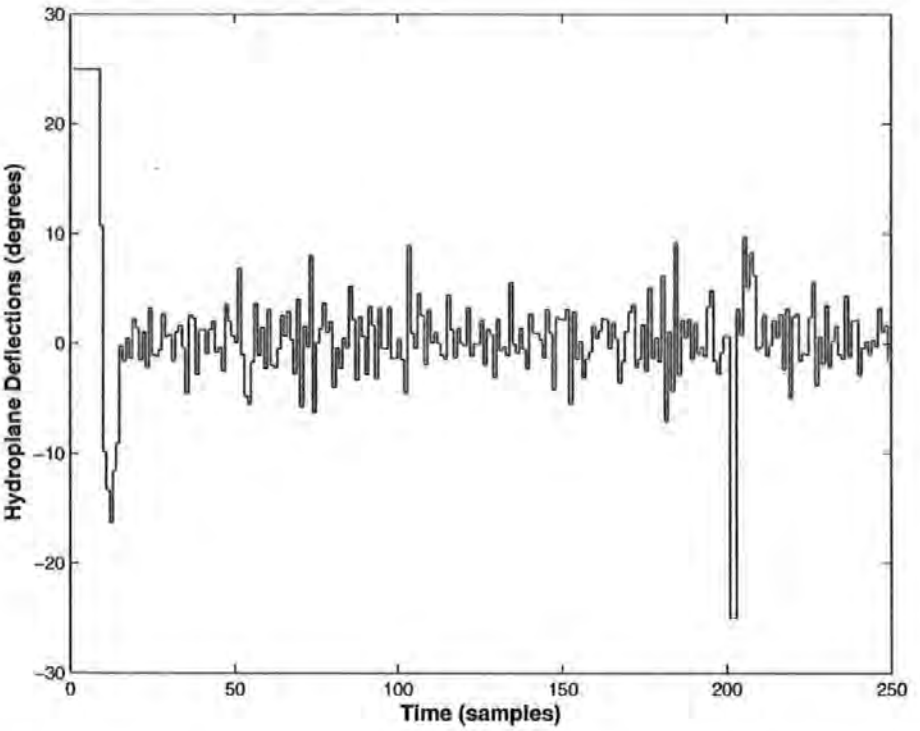


Figure 6.44: Reduced hydroplane deflections needed to track the specified depth (altitude) commands from the vision system

6.8 Concluding Remarks

The control of an underwater vehicle has always been a challenging task and is certainly an important and necessary feature of any underwater vehicle. The controller is responsible to keep the vehicle on specified course where the input to the autopilot is obtained from the onboard guidance system which generate suitable trajectories to be followed by the vehicle. This chapter has developed a discrete-time LQG/LTR based autopilot for the *Hammerhead* AUV. The controller has been designed using the yaw and depth channel models identified in Chapter 4. In addition, a model has been borrowed from literature to simulate the hybrid guidance and control system. The LQG/LTR is a robust optimal control methodology which is a combination of an LQR state feedback regulator and Kalman filter. The Kalman filter acts as an observer and provide estimates of the unmeasured and noisy states to the LQR as if the states are actually available.

A number of simulation results are presented in this chapter based on simple guidance schemes such as LOS and waypoint following. The performance of the control system is evaluated with and without the presence of disturbances. It is demonstrated

through simulations that the controller is quite robust for a diverse range of situations. Novel integrated guidance and control systems are developed based on the guidance laws proposed in Chapter 5. The results are found to be quite promising and are expected to add a fresh contribution to the underwater research literature. A depth autopilot is also developed and validated through simulation results. Finally, the depth controller has been shown to provide exceptional control of vehicle altitude even if the guidance commands are uncertain. The next chapter presents the formulation of a model based predictive controller combined with AI techniques such as genetic algorithms and fuzzy logic. The controller performance will be assessed against the LQG/LTR controller developed in this chapter.

Chapter 7

Model Predictive Control of *Hammerhead*

The main focus of this chapter is to develop a model predictive controller (MPC) for the *Hammerhead* AUV. MPC can provide robust control for processes with variable gain dynamics, multivariable interaction and constraints. The conventional MPC assumes a quadratic cost function and an optimization method such as quadratic programming (QP) to determine the optimum input to the process. Two variations are considered to the standard MPC problem for implementing in *Hammerhead*. In the first approach, genetic algorithms (GAs) are employed to find the optimal input by minimising the traditional quadratic cost error function. Next, the quadratic objective function is replaced by a performance index based on fuzzy membership functions. The advantages of both schemes are outlined and simulation results are presented to evaluate the performance of the proposed techniques.

7.1 Introduction

The proportional integral and derivative (PID) controller has been the workhorse in the process industry for over forty years. PID-type controllers are routinely used in SISO applications with good results, however, success with this type of controller for multivariable systems has been limited. It is now recognised that the limitations of the PID-type controller can be traced to its characteristics since it came into existence on the basis of hardware realisability (Deshpande, 1989). With the evolution of digital computers, much better designs can be produced without any hardware considerations. This, in part, has spurred research and development to evolve better control strategies for process systems.

Another source of incentive for the development of better control system design procedures lies in the demand of the times. Processes today are much more complex, requiring high level of steady state optimization and good closed-loop control. A major source of complexity in process plants is the existence of interacting multivariable systems and the presence of constraints. In order to be successful, any control system must anticipate constraint violations and correct for them in an efficient way. The usual practice in process control is to ignore the constraint issue at the design stage and then handle it in an *ad hoc* way during the implementation. MPC provides the only methodology to handle constraints in a systematic way during the design and implementation of the controller. Moreover, in its general form MPC is not restricted in terms of the model type and order, objective function and constraint functionality. For these reasons, it is the only technique that currently can reflect most directly the many performance criteria of relevance to the process industry hence claiming over 4600 successful applications to date in the industry (Lennox *et al.*, 2004).

The computation of control moves in MPC involves the solution of a constrained optimization problem where analytical solution is usually not available. These computations are obtained using numerical solution of the optimization problem at each sampling time, which necessitates the use of efficient optimization techniques. QP is generally used to solve the MPC online optimization problem. Herein, two modifications to the standard predictive control problem have been considered. The first scheme replaces the conventional optimizer by a GA. The proposed technique has proven to be a robust search methodology that requires little information to explore effectively in a large or poorly understood search space. A genetic search progresses through a population of points in contrast to the single point of focus of most optimization algorithms. In addition, the use of GAs allow the utilisation of any type of objective function and also has the capability to deal with any type of plant model and process constraints thereby generalising a range of MPC technologies which are distinguished on the basis of a process model and an objective function. The second modification proposed in an MPC control problem is to replace the quadratic cost function with a fuzzy objective function and a GA is employed to solve the resulting nonlinear optimization problem. The fuzzy objective function is intuitive because of its resemblance with human decision making by exploiting expertise knowledge. Additionally, the soft and hard constraints are automatically implemented in the fuzzy membership functions and no weighting matrices are needed to scale the various cost function parameters.

The next section describes the traditional predictive control problem followed by a succinct discussion on GAs in Section 7.3. Whilst section 7.4 outlines the GA-based predictive control problem formulation using a quadratic objective function and presents simulation results on *Hammerhead* yaw and depth control. Finally, the fuzzy GA-MPC problem is formulated in Section 7.6 and simulation results are illustrated pertaining to *Hammerhead* control in the horizontal and vertical planes to validate the concept.

7.2 Conventional Model Predictive Control

MPC refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimize the future behaviour of a plant. Originally developed to meet the specialised control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace and metallurgy (Qin and Badgewell, 2000), to name but a few.

The development of MPC can be traced back to 1978 after the publication of the paper by Richalet *et al.* (1978). They named their algorithm model predictive heuristic control (MPHC) and it was successfully applied to a fluid catalytic cracking unit main fractionator column, a power plant steam generator and a poly-vinyl chloride plant. Then Cutler and Ramaker from Shell Oil Company in 1979 and 1980 developed their own independent MPC technology referred to as dynamic matrix control (DMC) (Cutler and Ramaker, 1980). Simulation results are shown from a furnace temperature control application to demonstrate improved control quality. However, another form of MPC called generalised predictive control (GPC) devised by Clarke *et al.* (1987*a*), Clarke *et al.* (1987*b*), is employed in this paper. The fundamental difference between all these techniques is the type of model used and the cost function being optimized.

As noted above, the success MPC is enjoying is attributed to the fact that it was developed in an industry, by the industry and for the industry. A good account of MPC technology from the past to the future has been reviewed by Morari and Lee (1999), while a comparison between both theoretical and practical aspects of MPC has been undertaken by Carlos *et al.* (1989). For the interested reader, several other

useful references on MPC can be found (Soeterboek, 1992; Richalet, 1993; Clarke, 1994; Rawlings, 2000; Maciejowski, 2002) .

The process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the output of the process can be used. Furthermore, if the plant is subject to disturbances, a disturbance or noise model can be added to the process model. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used. Typically the criterion is the difference between the predicted process output and the desired reference trajectory. A simple error criterion function is given by Equation 7.1 as

$$J = \sum_{i=1}^{H_p} [\hat{y}(k+i) - w(k+i)]^2 \tag{7.1}$$

where $\hat{y}$ is the predicted process output, w is the reference trajectory, and H_p is the prediction horizon or output horizon. The structure of a standard MPC problem is depicted in Figure 7.1.

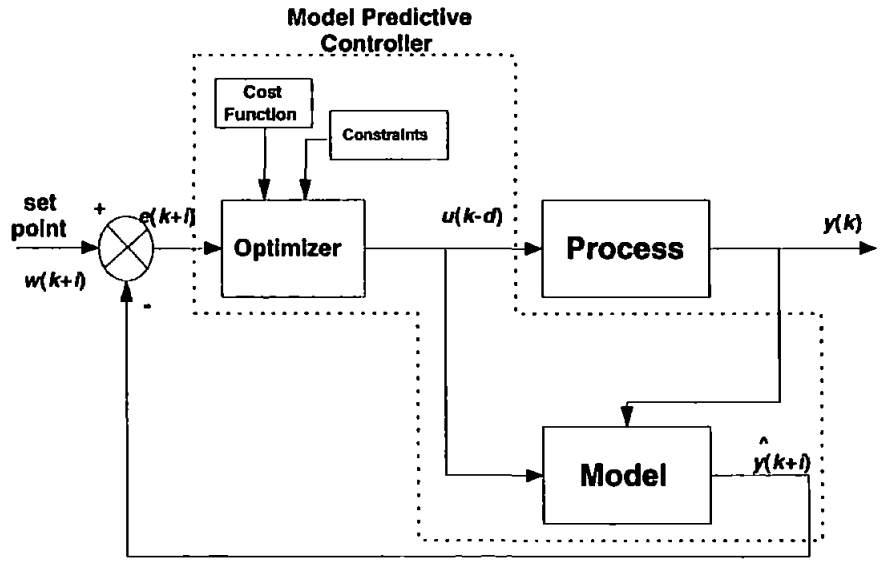


Figure 7.1: Structure of a conventional model predictive controller

As can be seen, the feedback mechanism is different from most standard control techniques. The model of the process is updated with actual measurements from the plant and the error e between the predicted model output and reference w is minimised by the optimizer over H_p. If there is no model mismatch i.e. the model is identical to the process and there are no disturbances and constraints, the process

will track the reference trajectory exactly on the sampling instants.

The MPC algorithm consists of the following three steps.

1. Explicit use of a model to predict the process output along a future time horizon (Prediction Horizon).
2. Calculation of a control sequence along a future time horizon (Control Horizon, H_c), to optimize a performance index.
3. A receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated as shown in Figure 7.2.

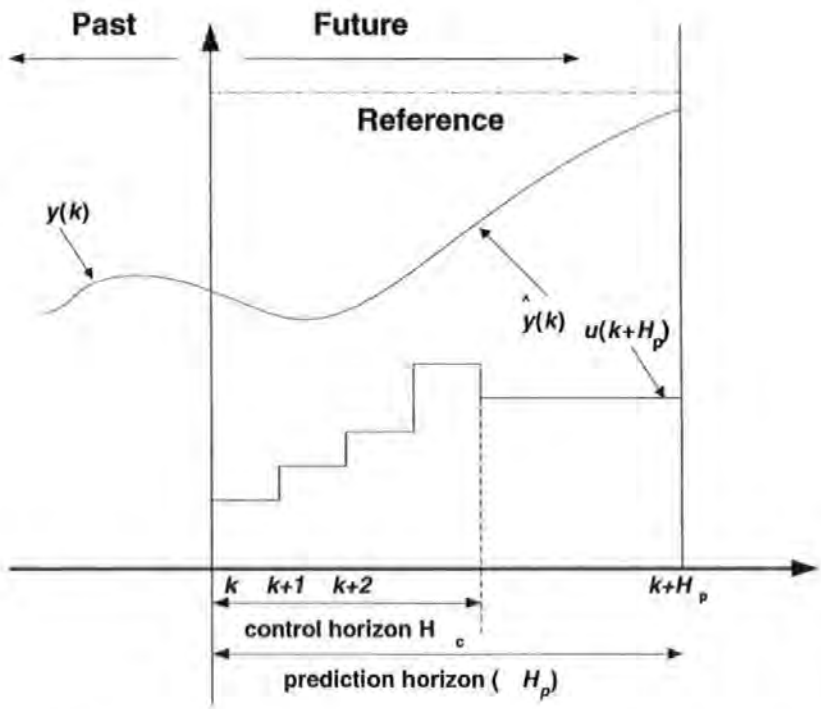


Figure 7.2: Predicted output and the corresponding optimum input over a horizon H_p, where $u(k)$ is the optimum input, $\hat{y}(k)$ represents the model prediction, and $y(k)$ is the process output

The selection of MPC to control an AUV is attributed to several factors. Some of them are listed below.

- The concept is equally applicable to SISO as well as multi-input, multi-output systems (MIMO).
- MPC can be applied to linear and nonlinear systems.
- It can handle constraints in a systematic way during the controller design.
- The controller is not fixed and is designed at every sampling instant based on actual sensor measurements so disturbances can easily be dealt with as compared to fixed gain controllers.
- The MPC success in the process industry provides a major impetus to transfer this technology into the marine sector. To the author's knowledge, there is no reported application of MPC in a marine vehicle (underwater and surface alike) to date thus providing sound motivation to experiment the technology in an AUV.

The next section describes a simple GA which is employed here to optimize the performance index.

7.3 Genetic Algorithms

GAs inspired by Darwinian theory, are powerful non-deterministic iterative search heuristics. GAs operate on a population consisting of encoded strings where each string represents a solution. The crossover operator is used on these strings to obtain new solutions, which inherits the good and bad properties of their parent solutions. Each solution has a fitness value and solutions having higher fitness values are most likely to survive for the next generation. The mutation operator is applied to produce new characteristics, which are not present in the parent solutions. The whole procedure is repeated until no further improvement is observed or run time exceeds to some threshold, (Sait and Youssef, 1999). The flowchart of a simple GA is depicted in Figure 7.3 and the operation of the GA is explained as follows.

To start the optimization, GAs use randomly produced initial solutions. This method is preferred when *a priori* knowledge about the problem is not available. After randomly generating the initial population of say N solutions, the GA uses the three genetic operators to yield N new solutions at each iteration. In the selection operation, each solution of the current population is evaluated by its fitness normally

represented by the value of some objective function and individuals with higher fitness value are selected in a probabilistic manner. Different selection methods such as roulette wheel selection and stochastic universal sampling can be used. The crossover operator works on pairs of selected solutions with certain crossover rate which is defined as the probability of applying crossover to a pair of selected solutions. There are many ways of defining this operator such as single point crossover, double point crossover, multi-point crossover etc. For example the single point crossover works on a binary string by determining a point randomly in the two strings and corresponding bits are swapped to generate two new solutions.

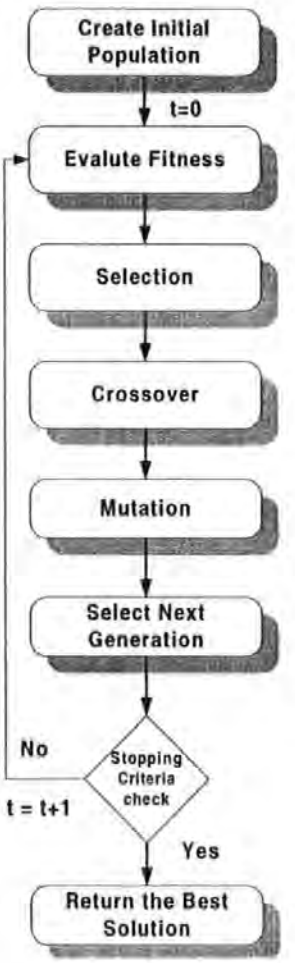


Figure 7.3: Flow chart of a simple genetic algorithm

Mutation is a random alteration with a small probability of the binary value of a string position. This operator prevents the GA from being trapped in local minima. The fitness evaluation unit in a GA acts as an interface between the GA and the

optimization problem. Information generated by this unit about the quality of different solutions is used by the selection operation in the GA. Next the stopping criteria must be decided. This may be the case when there is no significant improvement in maximum fitness or the maximum allowable time (number of iterations) is passed. At the end of the algorithm, the best solution found so far is returned.

The key advantages of using a GA in an MPC framework is its ability to handle various forms of objective functions, process models, and the ease through which constraints are implemented. Both linear and nonlinear models can be used with this approach without having to consider the local minima problem as is normally encountered in most linear optimization routines.

Section 7.4 outlines problem formulation of MPC when GA is employed as an optimization technique.

7.4 GA-Based Model Predictive Control

The genetic-based control algorithm for an AUV model is depicted in Figure 7.4. The algorithm proposed by Duwaish and Naeem (2001) was successfully simulated on a control valve and a heat exchanger processes which have been identified as nonlinear SISO Wiener and Hammerstein models respectively. The GA-based controller uses the process model to search for the control moves, which satisfy the process constraints and optimizes a cost function. One of the distinct advantages of using GA is the possibility of employing various objective functions and the ability to deal with any type of process model and constraints, thus generalising a range of MPC technologies where each of them is defined on a fixed set of process model and objective function. In this section, the conventional quadratic objective function is minimised to evaluate the control inputs necessary to track a reference trajectory and is given by Equation 7.2.

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) \quad + \quad \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \tag{7.2}$$

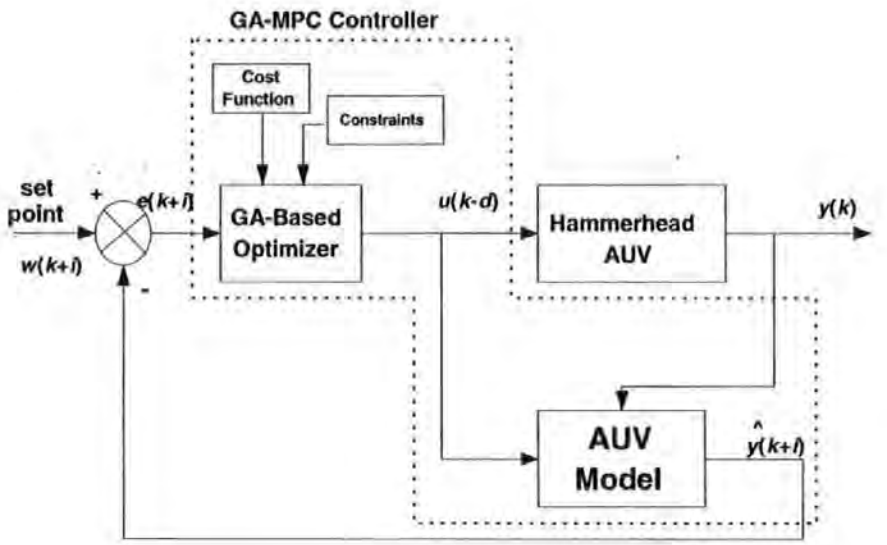


Figure 7.4: Genetic algorithm based model predictive controller

subject to input constraints

$$u^{-}_{constraint} \leq u(k) \leq u^{+}_{constraint}$$
$$\Delta u^{-}_{constraint} \leq \Delta u(k) \leq \Delta u^{+}_{constraint}$$

where the positive and negative signs represent the upper and lower constrained limits respectively. Q is the weighting scalar on the prediction error given by

$$e(k) = \hat{y}(k) - w(k) \tag{7.3}$$

where $w(k)$ is the reference or the desired setpoint. R and S are weights on the change in input Δu and magnitude of input u respectively. The following steps describe the operation of the GA-based MPC algorithm and the flow chart of the algorithm is shown in Figure 7.5. At every sample time k

1. Evaluate process outputs using the process model.

2. Use a GA search to find the optimal control moves which optimize the cost function and satisfy process constraints. This can be accomplished as follows.

   (a) generate a set of random possible control moves. The control moves or population consists of real values which is reasonable in a real world environment. The width of the population represents H_p and is doubled for every input added to the process.

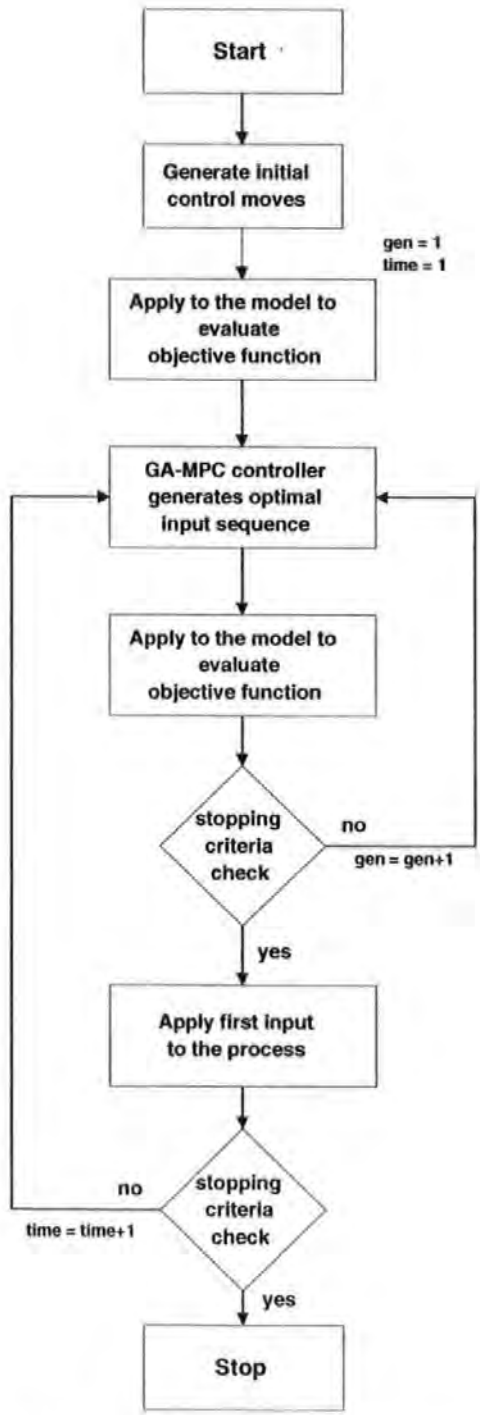


Figure 7.5: Flow chart of the GA-based MPC

(b) find the corresponding process outputs for all possible control moves using the process model over H_p.

(c) evaluate the fitness of each solution using the cost function and the process constraints. The fitness function used here is given by

$$fitness = \frac{1}{1+J} \tag{7.4}$$

where J is the cost function given by Equation 7.2.

(d) apply the genetic operators (selection, crossover and mutation) to produce new generation of possible solutions. Herein, single point crossover is used for parents mating whereas roulette wheel and stochastic universal sampling are employed for selection procedure.

(e) repeat until predefined number of generations is reached and thus the optimal control moves are determined.

3. Apply the optimal control moves generated in step 2 to the process.

4. Repeat steps 1 to 3 for time step $k+1$.

7.4.1 Constraints formulation

Constraints represent limitations on different physical quantities involved in a process. For instance, the input or output of a certain process is restricted beyond a specified value due to economical or environmental reasons or the input cannot be changed abruptly due to the hardware dynamics. One of the most powerful and distinguishing features of MPC is its ability to handle constraints in a natural way during the controller design at every sample time. Generally, two types of constraints are considered in controller design. Soft constraints are employed in the cost function as a penalty factor and can be violated to fulfil some other criteria. On the other hand, hard constraints represent physical limitations on actuators and cannot be violated. Here, hard constraints are placed on the input and rate of change of input variables. In this case, since the population in a GA represents the input variable, therefore, constraints are implemented by generating random initial population in the desired range i.e.,

$$u^{-}_{constraint} \leq u \leq u^{+}_{constraint}$$

where the superscripts $-$ and $+$ represents the lower and upper constraint limits respectively. Soft constraints on the input and rate of change of input are realised as penalty factors in the objective function in Equation 7.2. To implement the hard rate constraints, input for current time instant, k is compared with the input of the previous time instant, $k-1$. If the difference Δu of the two inputs is violating the constraint, i.e., if it is higher or lower than the desired range, it is adjusted to the limiting value accordingly by manipulating the input at the current time instant, k. Thus not allowing the rate of change of inputs to violate the constraint. Mathematically, it can be stated as

$$u(k) = u(k-1) + \Delta u^{-}_{constraint} \text{ if } u(k) - u(k-1) < \Delta u^{-}_{constraint}$$
$$u(k) = u(k-1) + \Delta u^{+}_{constraint} \text{ if } u(k) - u(k-1) > \Delta u^{+}_{constraint}$$

7.5 Simulation Results

In this section, the performance of the GA-MPC controller is evaluated using the *Hammerhead* models identified in Chapter 4. Likewise LQG/LTR controller explicated in Chapter 6, separate controllers are designed for the yaw and depth channels and various scenarios are assumed in simulation. The integration of guidance and control systems has also been undertaken where the guidance laws considered are the simple LOS guidance and waypoint following. In addition, guidance systems developed in Chapter 5 for cable tracking are also integrated with the GA-MPC autopilot. Simulation results are presented and the performance is compared with the LQG/LTR autopilot.

7.5.1 Heading control

The rudder-yaw channel of *Hammerhead* AUV has been extracted using SI techniques on trials data and is given by Equation 4.19. The input to this channel are the rudder deflections and output is the AUV heading angle. This model has a sampling period of 1 second and contains a pole on the unit circle which simulates a ramp response for a step input. There are physical limitations of $\pm 22.5°$ on the rudder actuator, however, hard constraints of $\pm 20°$ are imposed on the rudder movement through software in all simulations in order to avoid actuator saturation and hence any nonlinear behaviour. Initiating from an arbitrary angle, the goal is to attain a certain yaw angle with minimum control energy expense and maximum accuracy. The GA-MPC algorithm

is first simulated for a simple LOS guidance system where the vehicle is required to follow a LOS angle λ between two points (x_1, y_1) and (x_2, y_2) and is given by

$$\lambda = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{7.5}$$

The LOS angle to be followed is 200° measured with respect to the magnetic north whereas the initial vehicle orientation is arbitrary and assumed to be 50°. The weighting matrices and controller parameters used in the simulation for set point tracking are provided in Table 7.1.

Controller Parameters	Single set point tracking	Multiple set points tracking
Q	1	1
R	0.0	0.0
S	0.0	0.0
H_p	10	10
H_c	1	1
Mutation prob.	0.005	0.008
Crossover prob.	0.2	0.1
No. of generations	1	10
Population size	100	100
Insertion rate	0.4	0.25

Table 7.1: GA-MPC tuning parameters and weighting matrices for single and multiple LOS tracking missions

The resulting closed loop performance for a step change in AUV heading angle is depicted in Figure 7.6 showing that the vehicle is closely following the LOS with zero steady state error and minimum overshoot. The vehicle approaches the LOS angle swiftly and smoothly with a settling time of approximately 40 samples. The input rudder deflections are also shown in Figure 7.7. There are no hard rate constraints for this simulation therefore some large movements could be expected in the input. Clearly, the rudder movement stays within the specified boundaries and control energy expenditure is minimum. Comparing the performance with the LQG/LTR, the rate of change of vehicle's heading is approximately the same, however, the rudder movement is less active in this case.

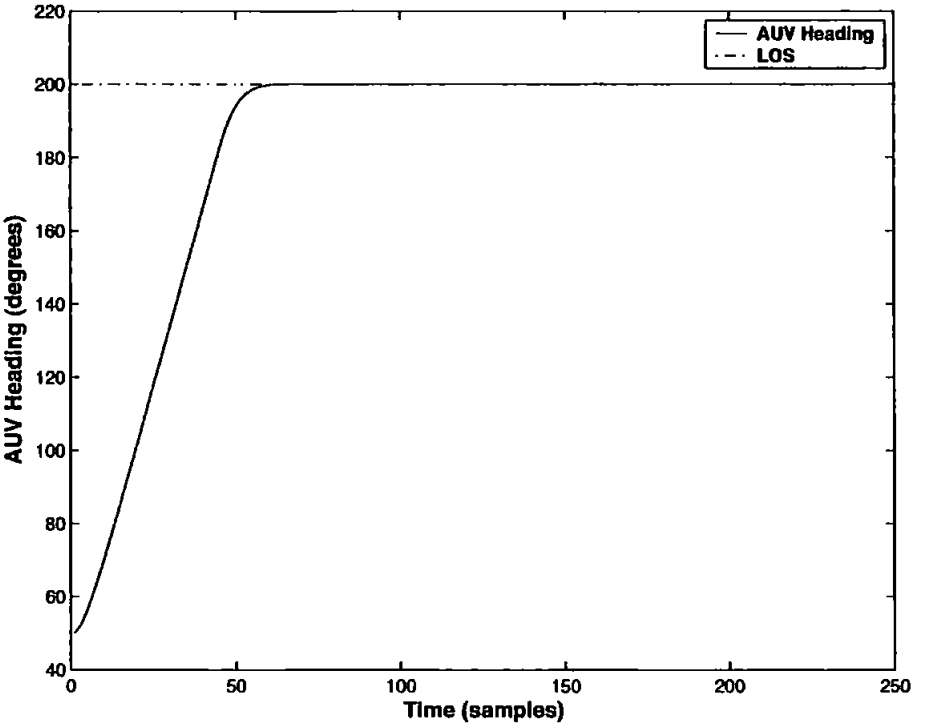


Figure 7.6: *Hammerhead* AUV closely following the LOS angle using a GA-MPC heading autopilot

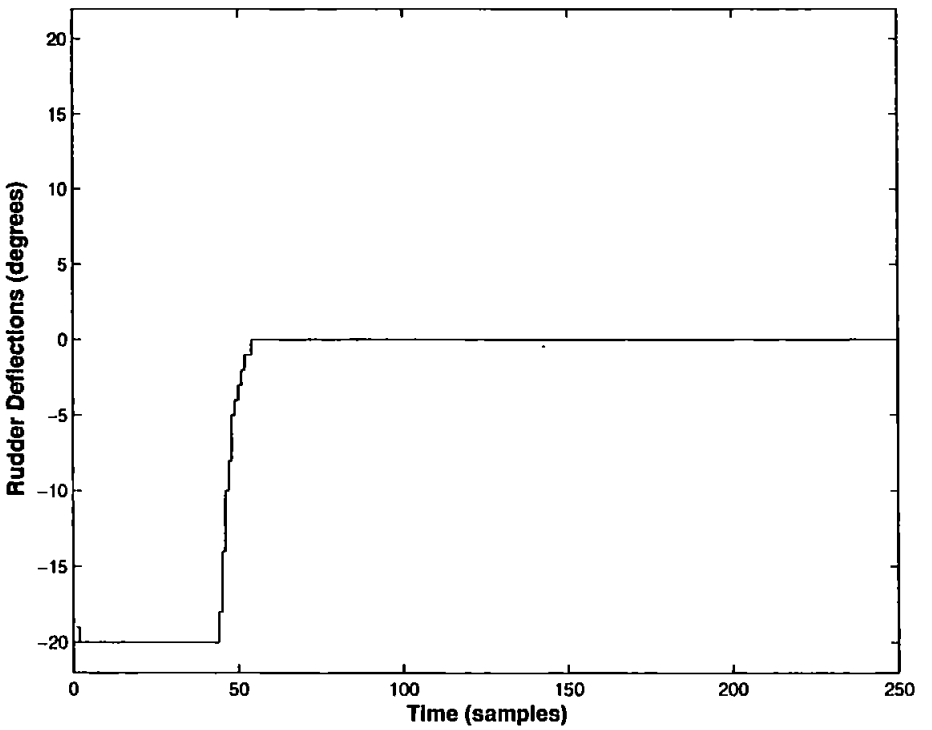


Figure 7.7: Input rudder deflections generated by the GA-MPC autopilot to follow the LOS angle

Please note that in practice, the *Hammerhead* rudder movement could not be controlled precisely. This is because the mechanical assembly in the vehicle has been retained from a 1960s made DMT. A minimum rate of change of input constraint has thus been imposed on the rudder movement for real time implementation of the controller in Chapter 8 and a feedback mechanism is also added for resetting purpose. Next, the algorithm is simulated to track several LOS angles during the course of a mission which is a typical case in a way point following scenario. Four step changes in heading angle are introduced at multiples of 75 sample times where the desired headings are 50°, 335°, 25° and 100°. The step change in heading from 50° to 335° (or −25°) has been chosen deliberately to show that the vehicle follows only the shortest path towards the desired heading as depicted in Figure 6.5.

The GA tuning parameters and weighting matrices for the objective function are also listed in Table 7.1. The vehicle as seen from Figure 7.8 detects all set point changes from the guidance system and tracks the desired heading angles closely with small overshoot and no steady state error. Again, the rudder movement shown in Figure 7.9 is within the specified constraints and indicate the presence of change in the output regime at multiple of 75 sample times.

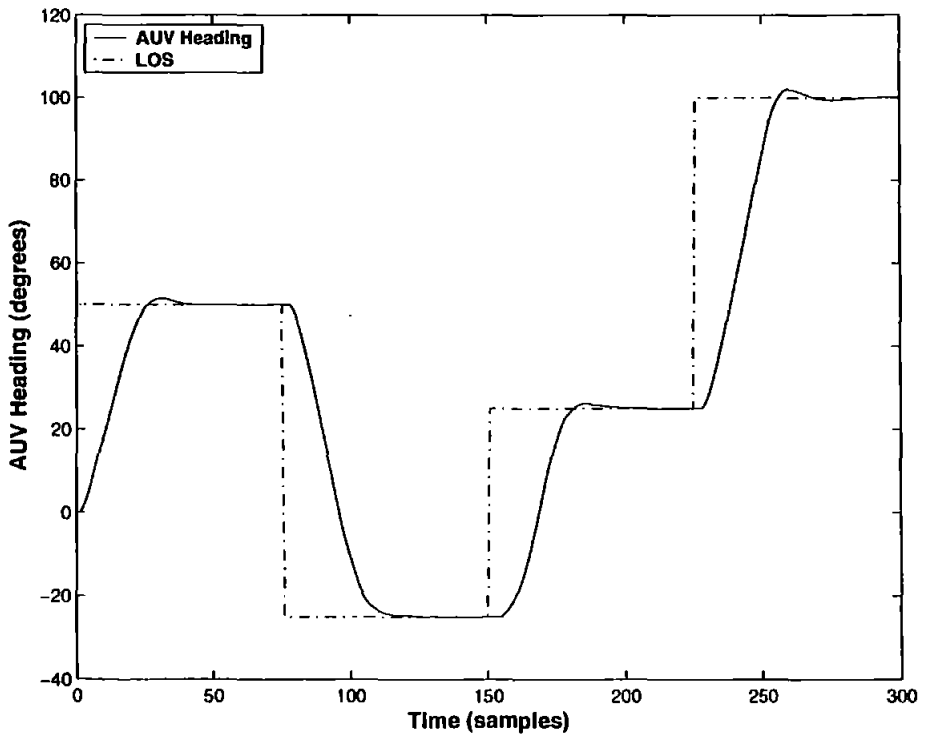


Figure 7.8: *Hammerhead* vehicle following the change in LOS guidance commands

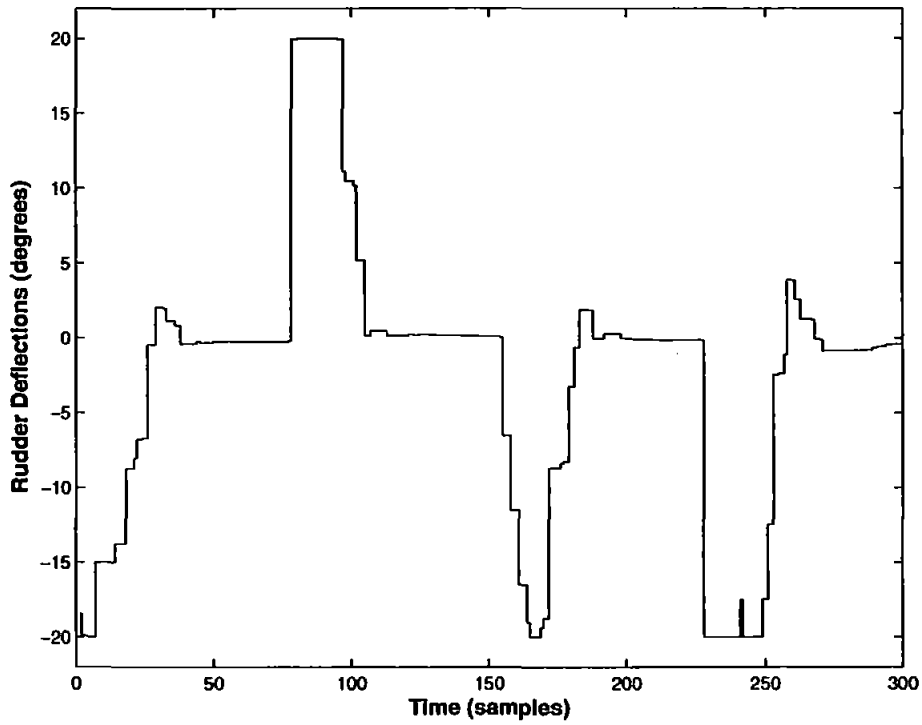


Figure 7.9: Optimal rudder deflections generated by the GA-MPC controller for several LOS angles

Proportional navigation guidance

The PNG law outlined in Chapter 5 is simulated here with an MPC controller in the loop. The results of the proposed algorithm have already been reported in a paper by Naeem *et al.* (2004*b*) based on *AUTOSUB* vehicle yaw dynamics. Herein, simulation results for a cable tracking mission are presented employing a *Hammerhead* rudder-yaw channel model given by Equation 4.19. An integration of the PNG law with LQG/LTR controller has been undertaken in Chapter 6 based on *Hammerhead* yaw dynamics and results demonstrate excellent cable tracking characteristics. Details of the proposed guidance law can be obtained from Chapter 5. In short, the proposed guidance law generate commands that are proportional to the LOS angle. Figure 5.2 depicted the Simulink block diagram of the proposed guidance system. The variable "theta1" represents the LOS angle which acts as an input to the control system. The AUV heading response denoted by "auv_heading" is used by the guidance system along with the vehicle velocity to update the desired LOS angle. The design of guidance and control systems has been taken on in Simulink and Matlab respectively and the information flow between them is shown in Figure 7.10.

Figure 7.10: Flow chart of the integration of MPC and PNG systems developed in Matlab/Simulink environment

The mission parameters are assumed to be the same as for the LQG/LTR autopilot. The vehicle is launched in the vicinity of $(0, 200)$ whereas the target FOC is $(0, 10)$ representing the seabed. A bias is introduced, as the guidance system suggests, after descending to a depth of $10m$ above the sea bed. The controller parameters used in the simulation are summarised in Table 7.2.

Controller Parameters	Value
Q	1
R	0.0
S	0.0
H_p	7
H_c	1
Mutation prob.	0.005
Crossover prob.	0.7
No. of generations	10
Population size	100
Insertion rate	0.2

Table 7.2: GA-MPC tuning parameters for a cable tracking mission using a PNG law

Figure 7.11 depicts simulated tracking of an underwater cable using the integration of the proposed PNG and GA-MPC systems. The onboard sonar provides the necessary cable position information which is utilised by the guidance system to estimate the LOS angle which is to be followed. The introduction of $10m$ bias in the vehicle y-coordinate prevents the AUV to actually intercept the cable and hence monitor it at a close length. The *Hammerhead* trajectory obtained with an LQG/LTR controller is quite similar to the one generated with an MPC. The difference could however be observed from the heading plot in Figure 7.12. In this case, the vehicle dynamics are more obvious and there is a small lag between the desired and vehicle's heading in contrast to Figure 6.24. Moreover, the rudder deflections shown in Figure 7.13 are less aggressive than in Figure 6.25 for the LQG/LTR case where the rudder has a large spike from the positive to negative side which is absent here. Figure 7.13 suggests that the positive constraint is active for the first 20 samples when the vehicle was making a turn. However, the GA based controller is aware of this and hence never violated the actuator's physical saturation limits.

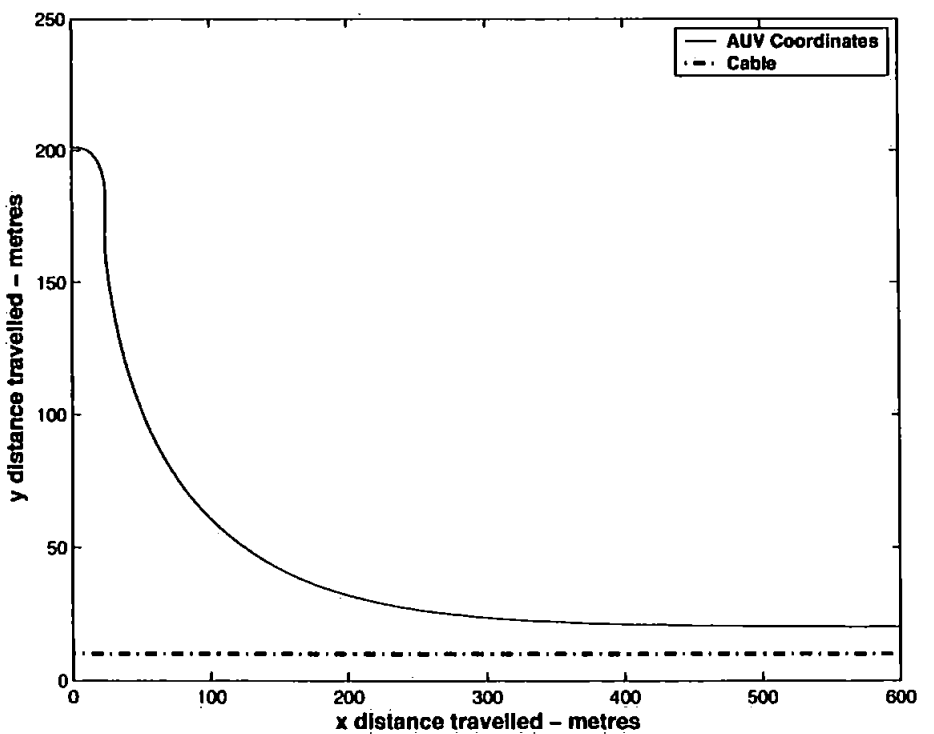


Figure 7.11: AUV and target position coordinates generated by the GA-MPC autopilot where the AUV is tracking the cable at a specified height

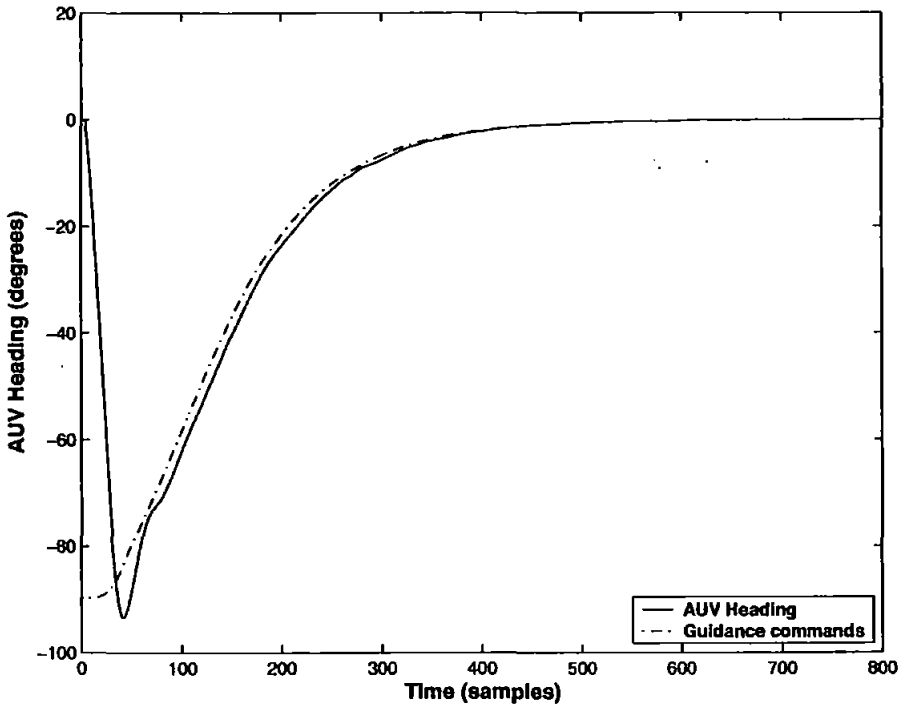


Figure 7.12: AUV heading controlled by the GA-MPC following closely the guidance commands generated by the PNG

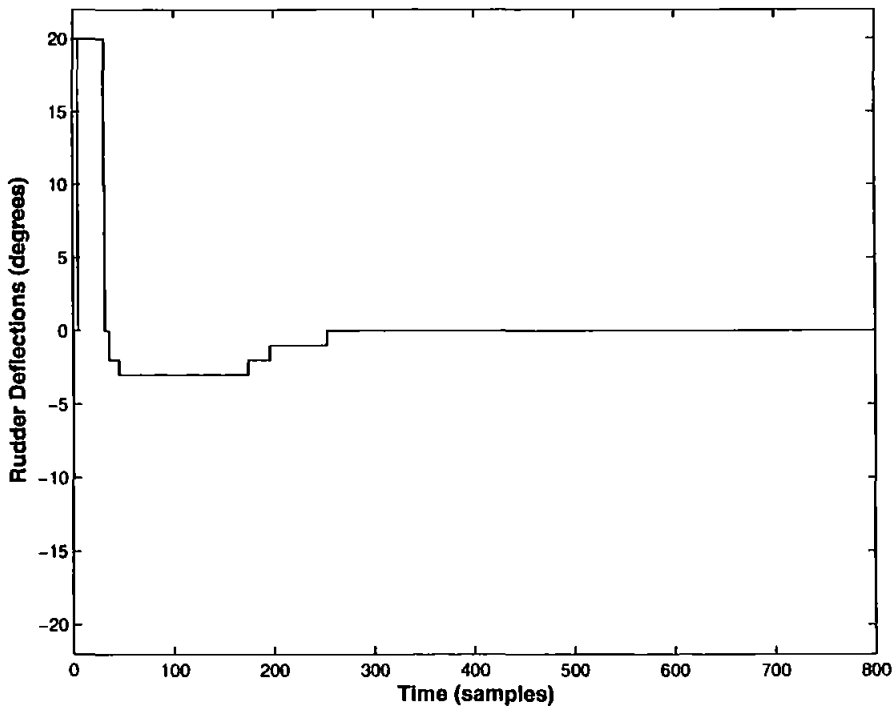


Figure 7.13: Rudder deflections evaluated by the GA-MPC controller needed to track the reference trajectory generated by the PNG

Waypoint following

Waypoint guidance scheme proposed by Healey and Lienard (1993) and reviewed in Chapter 2 has been simulated here with a GA-based MPC. The guidance law suggests to employ waypoints between the target and launching positions and to follow them to reach the destination. A COA is defined around the waypoint so that when the AUV enters this circle, the next waypoint is selected. The radius of the COA is recommended by Healey and Lienard (1993) to be at least twice the length of the vehicle.

In this section, the GA-MPC controller is simulated for waypoint following for the same cases considered in Chapter 6. The AUV is assumed to be manoeuvring near to the surface so that an onboard GPS provides the vehicle position coordinates in the earth-fixed frame of reference. The vehicle launching coordinates are $(0, 20)$ whereas the waypoints in the (x, y) plane are assumed to be those provided in Table 6.1. The next position is selected when the vehicle enters the COA of any given waypoint. The *Hammerhead* length is approximately three and a half metres therefore the radius of COA is taken to be $10m$ which is more than half of the AUV length. Table 7.3 provides the controller parameters used in the simulations.

Controller Parameters	Value
Q	1
R	0.0
S	0.0
H_p	7
H_c	1
Mutation prob.	0.05
Crossover prob.	0.1
No. of generations	10
Population size	100
Insertion rate	0.5

Table 7.3: GA-MPC tuning parameters for waypoint following with and without sea current disturbances

The algorithm is first simulated without considering any disturbances and the result is unveiled in Figure 7.14 showing the vehicle trajectory through the waypoints. The

affect of vehicle dynamics is evident from the figure, in particular, during turning manoeuvres. The AUV closely follows the ideal trajectory, however, it is much more vulnerable to sea current disturbances having a higher magnitude because of its slow dynamics. The MSE between the actual and ideal AUV trajectories is approximately $1.96m^2$. This was estimated at approximately $9.6m^2$ when LQG/LTR was used as an autopilot in Chapter 6. The rudder deflections generated by the controller is also shown in Figure 7.15 marking the waypoint changes since the vehicle has to manoeuvre in order to follow the updated waypoints. However, the controller never violates the actuator constraints. The vehicle heading angle is illustrated in Figure 7.16 showing the various LOS angles the AUV had to follow during the course of its mission between different waypoints.

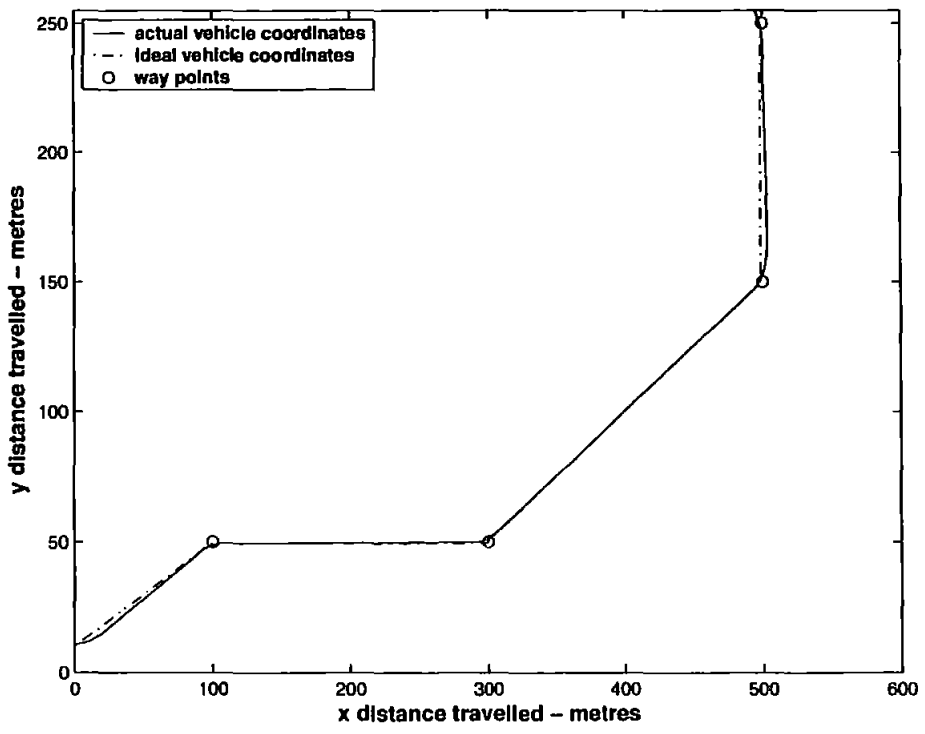


Figure 7.14: GA-MPC control of *Hammerhead* for waypoint following showing AUV and target position coordinates without sea current disturbance

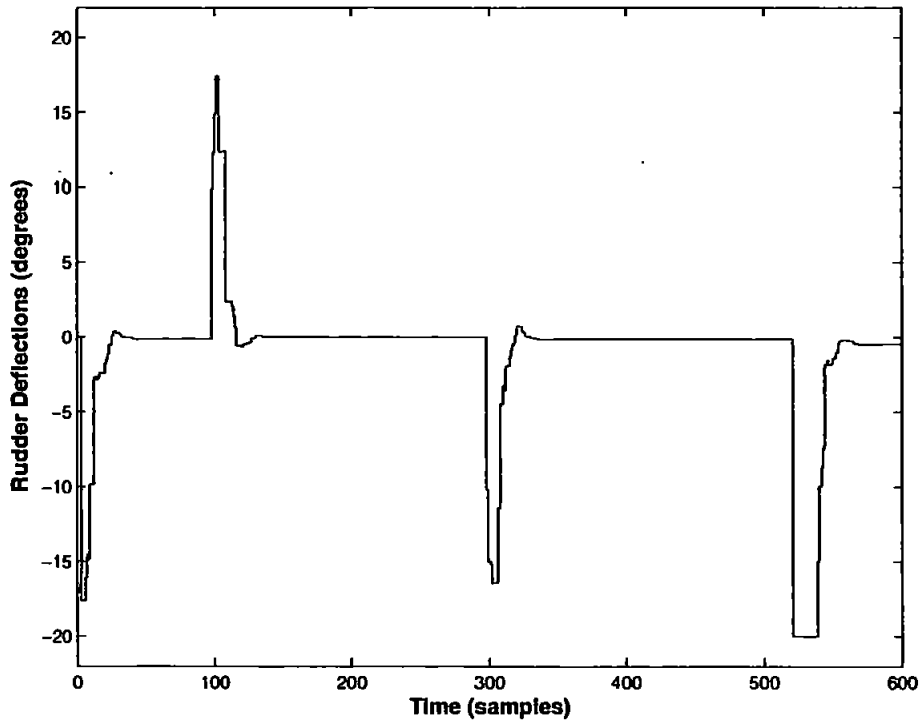


Figure 7.15: Rudder deflections generated by the GA-MPC autopilot needed to track the waypoints without sea current disturbance

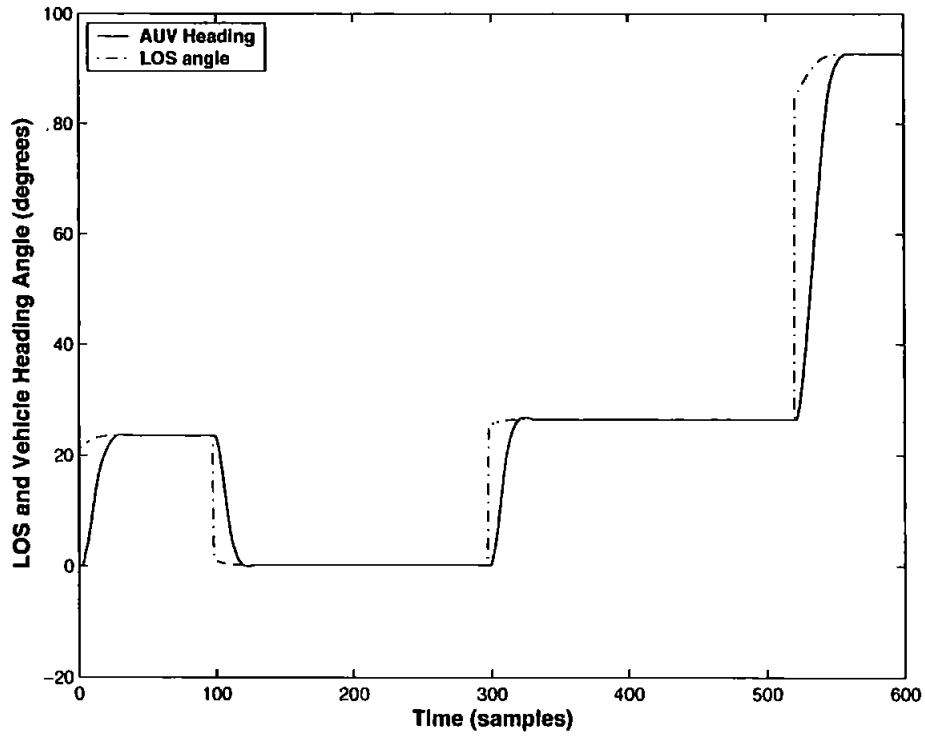


Figure 7.16: GA-MPC control of *Hammerhead* showing the vehicle heading and the corresponding LOS angles in waypoint following without any sea current disturbances

Finally, the controller is tested in the presence of sea currents where the disturbance model used is given by Equations 4.26 and 4.27. A sea current of 0.5 m/s (half the vehicle velocity) is assumed to be acting on the vehicle in the direction of positive y-axis as in Chapter 6. The response of *Hammerhead* by the addition of sea currents on waypoint following is shown in Figure 7.17. The disturbance is trying to force the vehicle off the track, but the controller is still able to cope with it and reaches the target waypoints. A maximum deviation of less than $50m$ is recorded when the vehicle is travelling in the positive x-direction. The sea currents favour the vehicle motion when moving in the direction of current from $(500, 150)$ to $(500, 250)$. The MSE is about $1049m^2$, much larger than the no disturbance case albeit the vehicle follows all the waypoints even in the worst case scenario of the sea current direction. The MSE in this case is quite high as compared to the one obtained using LQG/LTR. The rudder movement depicted in Figure 7.18 is also confined within the specified limits, however, it is lot more aggressive as compared to the no disturbance case. A statistical analysis reveals that the STD of rudder deflections in the presence of wave disturbance is about 9° while it is approximately half that value without any disturbance. Similar values were obtained in Chapter 6, when LQG/LTR controller was employed.

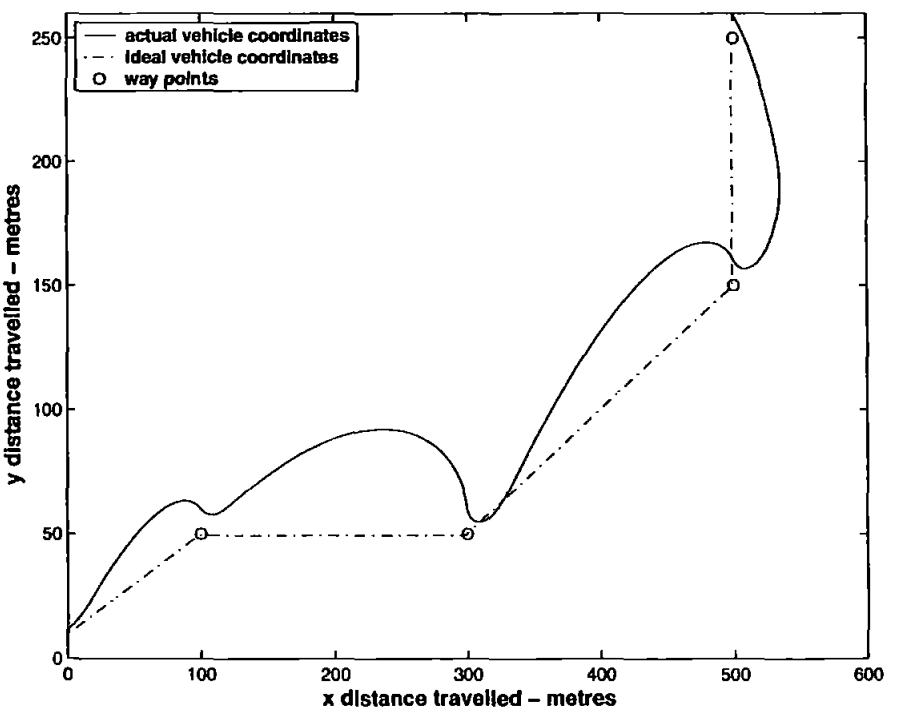


Figure 7.17: GA-MPC control of *Hammerhead* for waypoint following showing the AUV and target position coordinates with sea currents in the positive y-direction

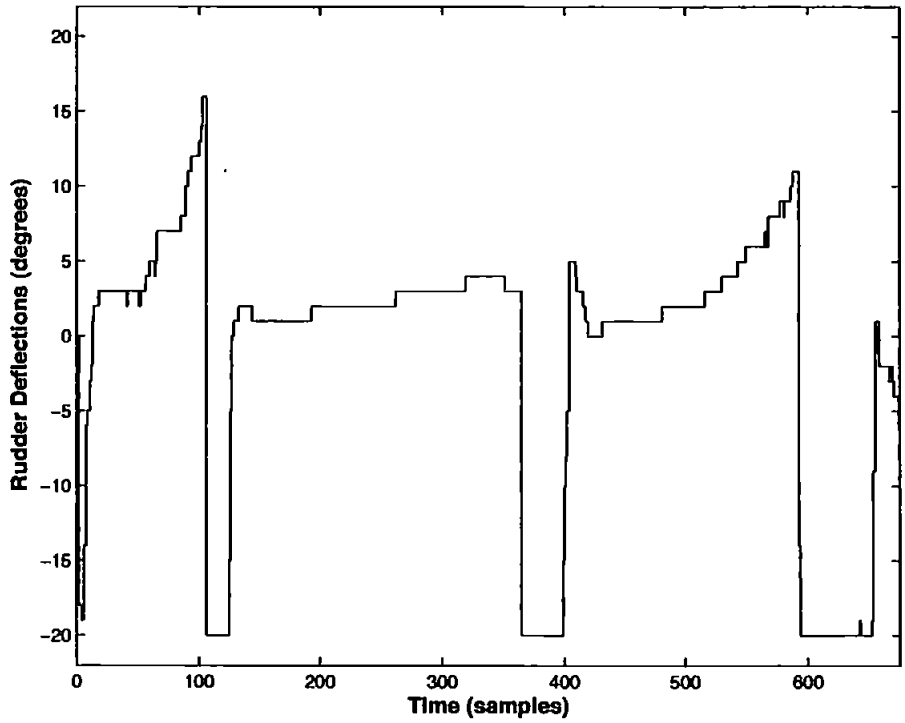


Figure 7.18: Rudder deflections generated by the GA-MPC autopilot needed to track the waypoints with sea currents in the positive y-direction

Figure 7.19 illustrates the impact of sea currents on the LOS angle. Since the vehicle position is continuously changing due to surface current forces, therefore the LOS angle is not constant as in no disturbance case. However, equipped with the GA-MPC controller, the vehicle is capable of following the desired yaw despite the presence of disturbances.

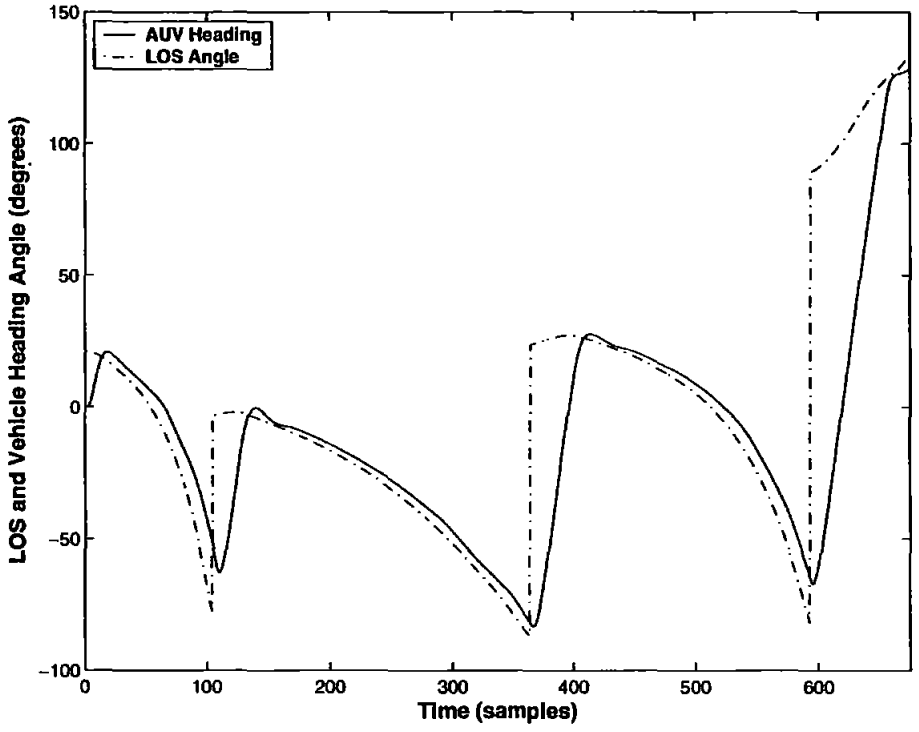


Figure 7.19: AUV heading angle and the corresponding LOS angles in a waypoint following mission with sea current in the positive y-direction

Hybrid guidance law

The hybrid guidance law proposed in Chapter 5 is integrated in this section with the GA-MPC autopilot. The hybrid guidance algorithm involves changing vehicle speed depending on the distance from the target. The complete problem formulation can be found in Section 5.2. Since the *Hammerhead* model cannot be used here, therefore an AUV model has been borrowed from literature which is represented in terms of vehicle velocity. The details of the model are given in Chapter 6, Section 6.7.1. Three separate models are obtained at velocities of 5, 7.5 and 10 knots and the resulting continuous time model is discretised at a sampling rate of $10Hz$. The model AUV is assumed to have a turning radius of $25m$ and the constraints on the rudder actuator are $\pm 25^0$. Controllers are developed one for each speed model which are switched on the basis of vehicle distance from the waypoint i.e., the known end of the cable. The mission parameters are the same as were assumed in Chapter 6 for the hybrid guidance simulation. There are four phases of the mission during which the vehicle speed is reduced as it crosses each phase boundary from midcourse to terminal and from terminal to tracking phase. The controller parameters tuned at three vehicle

velocities turns out to be the same for optimum performance and are supplied in Table 7.4.

Controller Parameters	Value
Q	1
R	0.0
S	0.0
H_p	15
H_c	1
Mutation prob.	0.008
Crossover prob.	0.1
No. of generations	10
Population size	75
Insertion rate	0.25

Table 7.4: GA-MPC tuning parameters for fixed and variable vehicle velocities in a hybrid guidance law

The closed loop response of the control system is illustrated in Figure 7.20 where the vehicle trajectory at a fixed speed is compared with the vehicle position at varying velocities. During the launching and midcourse phases, the vehicle was running at 10 knots (top speed) which was reduced to 7.5 knots on entering the COA during the terminal phase. Finally, the vehicle is slowed further down to 5 knots when within close proximity of the cable. The speed is then maintained at 5 knots throughout the tracking phase. Clearly, the vehicle trajectory at fixed speed has some oscillations before it runs in parallel with the cable. On the other hand, the trajectory is quite smooth when the vehicle velocity is varied and there are no oscillations present. A comparison of Figure 7.20 with Figure 6.30 reveals that the GA-MPC controller outperforms the LQG/LTR based autopilot. The vehicle's trajectory even at a fixed speed as shown is rather smooth as compared with Figure 6.30 which shows excessive and unwanted vehicle manoeuvres.

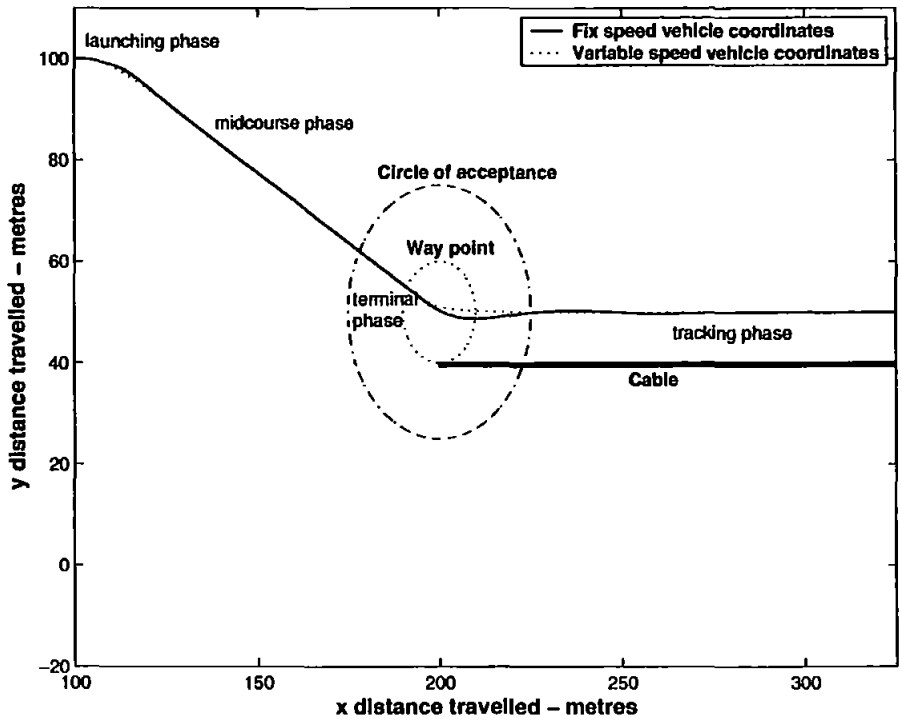


Figure 7.20: Simulation of the hybrid guidance law showing the AUV coordinates for fixed and variables speeds

The affect of changing vehicle velocity is more obvious from Figure 7.21 which shows the AUV heading angle for both cases. The vehicle orientation is fixed throughout the midcourse phase at $-29°$ and varies during the terminal and tracking phases. By reducing the velocity, the overshoots and oscillations can be greatly avoided as evident from the figure. The control surface deflection in a hybrid guidance strategy depicted in Figure 7.22 is slightly more aggressive than in normal mode as shown in Figure 7.23. However, it is contained within the specified constraints at all times. Again a comparison of the rudder movements between MPC and LQG/LTR clearly reveals that the performance of GA-MPC is outstanding. The peak rudder movement is not more than 10^0 as Figure 7.23 suggests whereas actuator saturation was frequently encountered in case of LQG/LTR.

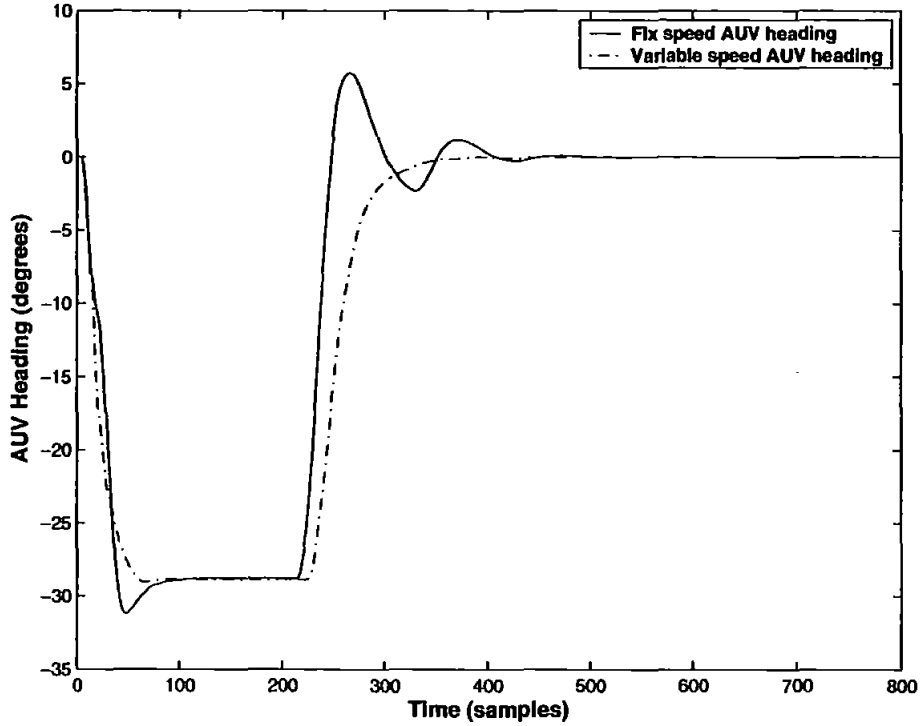


Figure 7.21: Simulation of the hybrid guidance law integrated with a GA-MPC autopilot showing the affects of changing the vehicle speed on heading angle

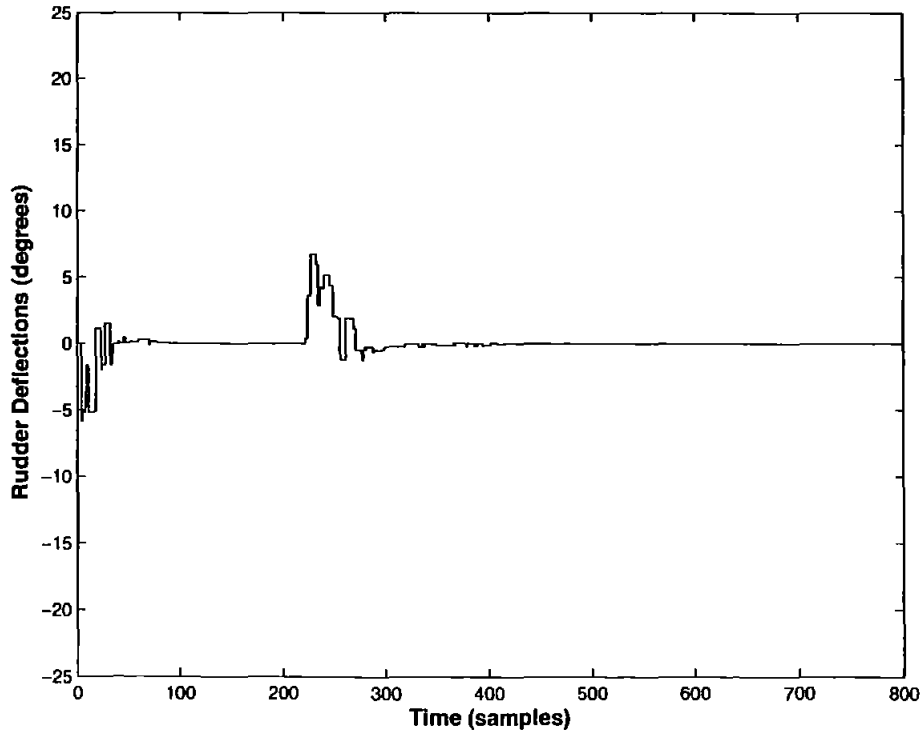


Figure 7.22: Rudder deflections generated by the GA-MPC controller with the vehicle velocity changing according to the hybrid guidance strategy

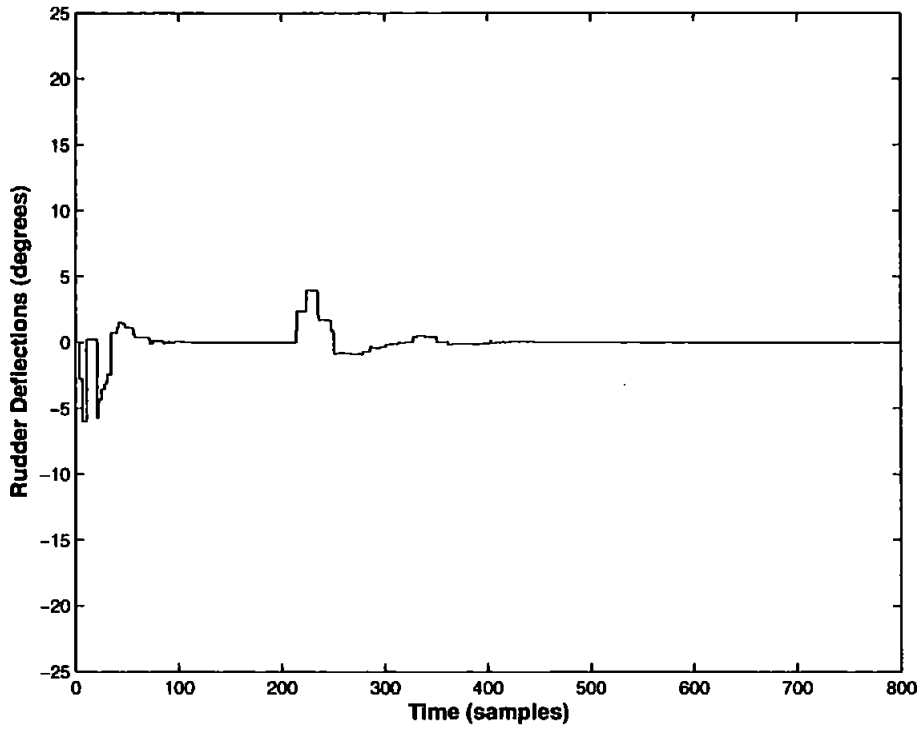


Figure 7.23: Rudder deflections generated by the GA-MPC autopilot with the vehicle manoeuvring at a constant velocity

7.5.2 Depth control

Having designed a GA-based MPC heading controller for *Hammerhead* using a quadratic objective function, attention is focused on developing a depth autopilot. The model used for controller design is given by Equation 4.23 with a sampling time of 1 second and identified from trials data using SI techniques. As addressed in Chapter 4, the vehicle shows cross coupling between the depth and yaw channels. However, since the umbilical was thought to be the most probable cause of this phenomenon which has been replaced by a wireless link, therefore it is envisaged that separate heading and depth models would be adequate for control purposes. Moreover, there is a strong cross coupling observed between the pitch and depth channels which is ignored for simplicity and for the sake of preliminary control system design and testing.

Similar to rudder-yaw channel model, the depth dynamics has a pole at the unit circle indicating a ramp output in response to a step input. In practice, the depth data is obtained either directly from a pressure transducer or from an inertial sensor onboard the AUV which provides acceleration in x, y and z dimensions and hence

can be integrated to obtained the position information in the dive plane. The *Hammerhead* AUV is kept slightly positively buoyant for ease of recovery, therefore it requires relatively large force from the control surfaces to dive which is generated by a positive hydroplane deflection of a greater magnitude. Once the vehicle completely submerges, small deflections are sufficient to control the depth.

The intent is to design a controller that is able to control the depth of the AUV as closely as possible despite the presence of external disturbances and modelling errors. The performance of the control strategy developed herein will eventually be assessed on the *Hammerhead* vehicle. The response to a step change in depth while the vehicle is on the surface is evaluated first. The next step is to simulate the response of *Hammerhead* for multiple step changes when the vehicle is at a certain depth level. The controller parameters including the weighting matrices, prediction and control horizons and GA variables are provided in Table 7.5.

Controller Parameters	Step response	Multistep response
Q	1	1
R	0.0	0.07
S	0.0	0.0
H_p	10	20
H_c	1	1
Mutation prob.	0.005	0.005
Crossover prob.	0.2	0.3
No. of generations	15	10
Population size	100	100
Insertion rate	0.5	0.5

Table 7.5: GA-MPC tuning parameters for depth step response simulations

The front canards movement is restricted to $\pm 25°$ which was thought to be adequate to control the vehicle in the vertical plane and was obtained through a series of rigorous in water experiments. Please note that there is no weight on the rate of change of input parameter in the objective function for the step response simulation. It is assumed that the vehicle is manouevring near the surface (zero depth) and is subjected to a depth command of $3m$. The *Hammerhead* response to a step change in depth when diving from the sea surface is shown in Figure 7.24.

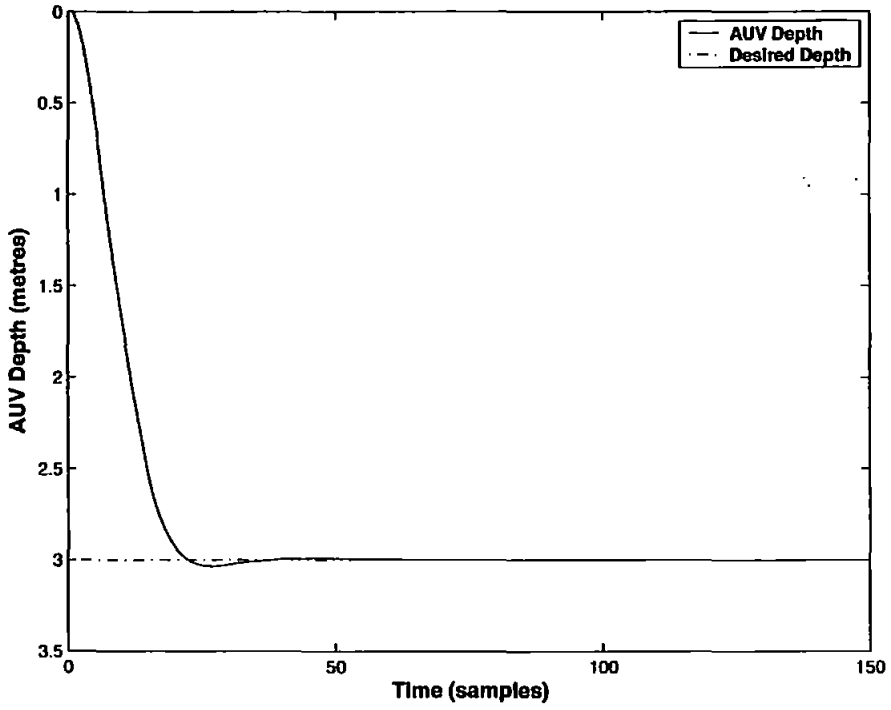


Figure 7.24: *Hammerhead* step response for a change in depth obtained by employing a GA-MPC depth autopilot

It took less than 25 sample times for the vehicle to attain the desired $3m$ of depth with little overshoot and no steady state error. The diving rate stays uniform as the vehicle approaches the desired level with precision. A quick comparison of results in Figure 7.24 and Figure 6.36 reveals that the LQG/LTR controller performance is better for depth control. The vehicle response in Figure 6.36 is rather quick with settling time less than 15 samples in contrast to 25 samples in this case. The disagreement can be explained by looking at the hydroplane deflection plot in Figure 7.25 and a comparison is made with Figure 6.37. The maximum control surface movement for the GA-MPC is approximately 13^0 as compared to $\pm 25^0$ for the LQG/LTR which explains the slower response time.

The simulations are carried out next for multiple step changes in depth initiating from the surface and when submerged. The vehicle response is also evaluated for a depth command in the upward direction which is facilitated by the positive vehicle buoyancy. In this case, the control effort required should be lower than needed for a depth demand towards the sea bed. The controller parameters are given in Table 7.5 where it should be noticed that there is a weighting factor introduced for the rate of

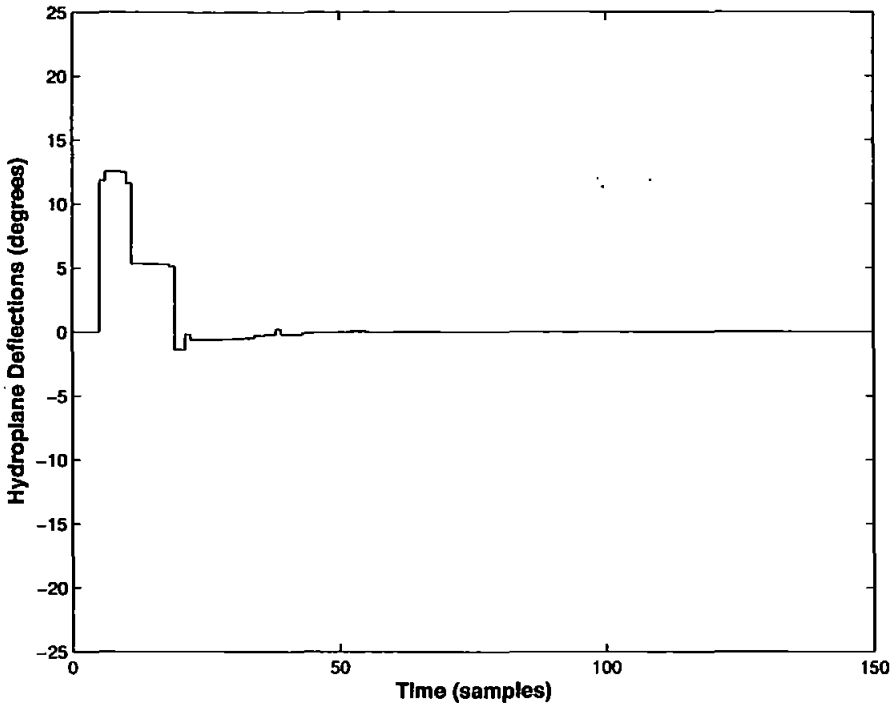


Figure 7.25: Hydroplane deflections generated by the controller to attain the desired step change in depth

change of control surface movement. The response of the closed loop system for a depth manifold is depicted in Figure 7.26. The *Hammerhead* response is nearly the same for a depth change of 3, 6 and 8 metres. This is followed by a set point change from 8 to 5 metres and it is apparent that the closed loop vehicle response is excellent without any overshoot and no steady state errors. The difference could however be noticed in Figure 7.27 showing the control demands required to attain and maintain a specified depth. The maximum positive deflection is 8 degrees which is lower than the previous case because of the rate of input weighting. A negative hydroplane movement of about 13 degrees is also observed when the vehicle is manouevring towards the sea surface. However, the control effort required is clearly quite low and stays within the specified constraints at all times. Similar reasoning could be made as mentioned before when the performance is compared with the LQG/LTR controller for a multistep response.

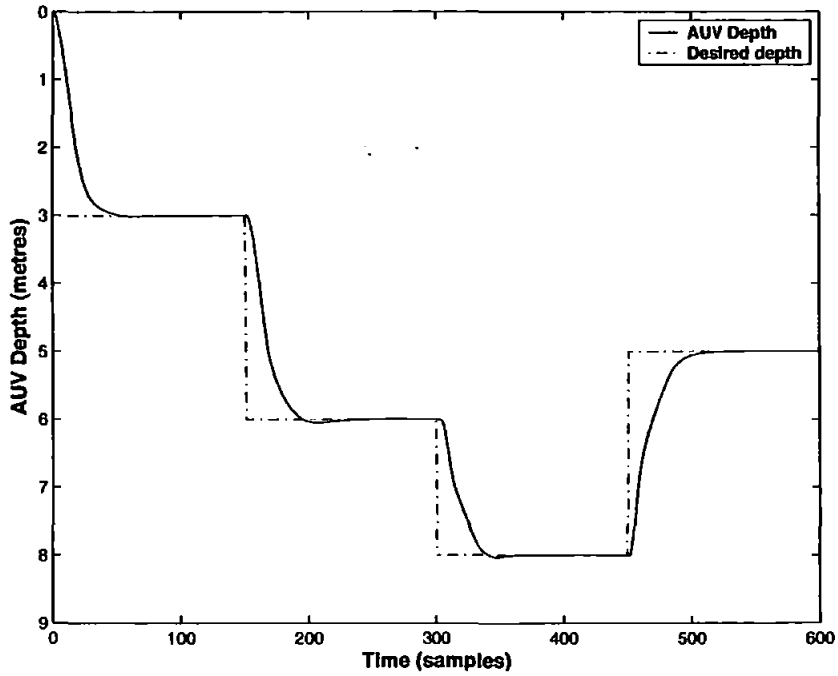


Figure 7.26: *Hammerhead* response to several step changes in depth command using a GA-MPC autopilot

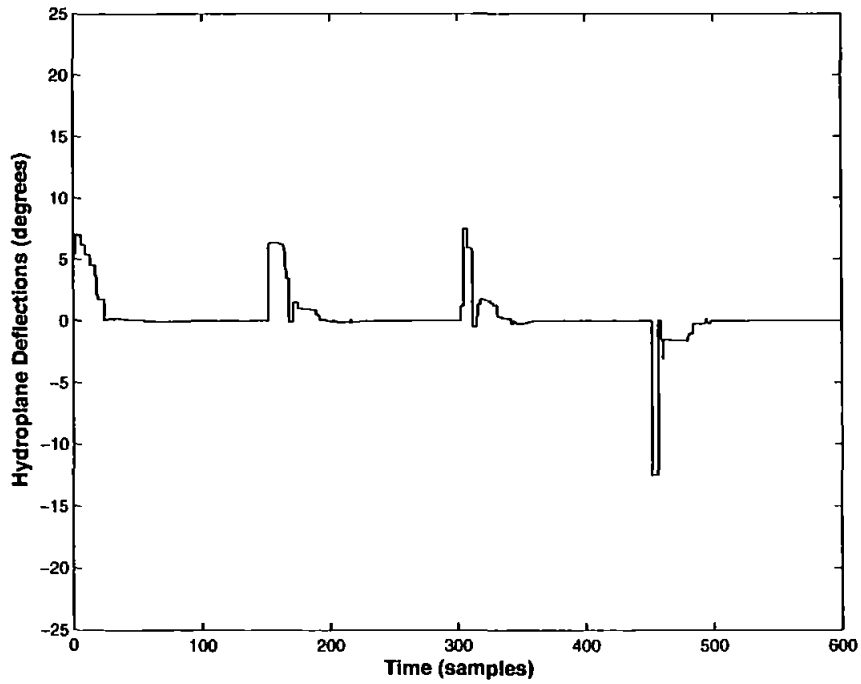


Figure 7.27: Hydroplane deflections generated by the GA-MPC depth controller for several changes in depth demand

Altitude control

Simulation results are now presented when the *Hammerhead* control system developed in this chapter is assumed to be integrated with an onboard vision system. Details of the integration can be found in Chapter 6. In short, the vision system could demands a specific altitude to maintain a certain image quality. The altitude information is then converted into desired depth data and the autopilot is simulated to control the depth of the vehicle. The results shown in this section have been published by Dalgleish *et al.* (2004). A normally distributed random noise of zero mean and variance 0.001 is added to the vision system output to simulate the uncertainty in the guidance commands. The *Hammerhead* heading is shown in Figure 7.28 illustrating the actual and desired AUV depths. It is obvious that the performance of the GA-MPC autopilot is far better than the LQG/LTR when a comparison is made with Figure 6.40. The result demonstrates excellent disturbance rejection properties of the controller as it tracks the desired set points while at the same time, ignoring the high frequency contaminations contained in the guidance signal. On the other hand, the LQG/LTR controller was tracking all the unwanted noisy movements in the input thus wasting precious control energy. Figure 7.29 depicts the hydroplane deflections which are rather small as compared to Figure 6.41 which shows excessive amount of energy being wasted. The plot in Figure 6.44 has smaller control surface deflections due to the increase in the magnitude of control weighting matrix R. However, the GA-MPC performance is still far more superior than the LQG/LTR autopilot.

7.6 GA-MPC Using Fuzzy Objective Function

So far in this chapter, a model based predictive controller has been explored employing a quadratic objective function. The resulting optimization problem was solved using a GA and simulation results for an AUV heading and depth control were presented for various scenarios to demonstrate the efficacy of the approach. The advantages of using a GA is its potential of solving complex nonlinear or constrained linear problems. In particular, in a GA-MPC framework, any type of linear and nonlinear models can be employed with or without the presence of constraints. Furthermore, the algorithm is generalised to take advantage of various other forms of objective functions and process constraints. The algorithm was shown to be simulated only for linear constrained systems, however, it is equally applicable to nonlinear, non-convex constrained plants.

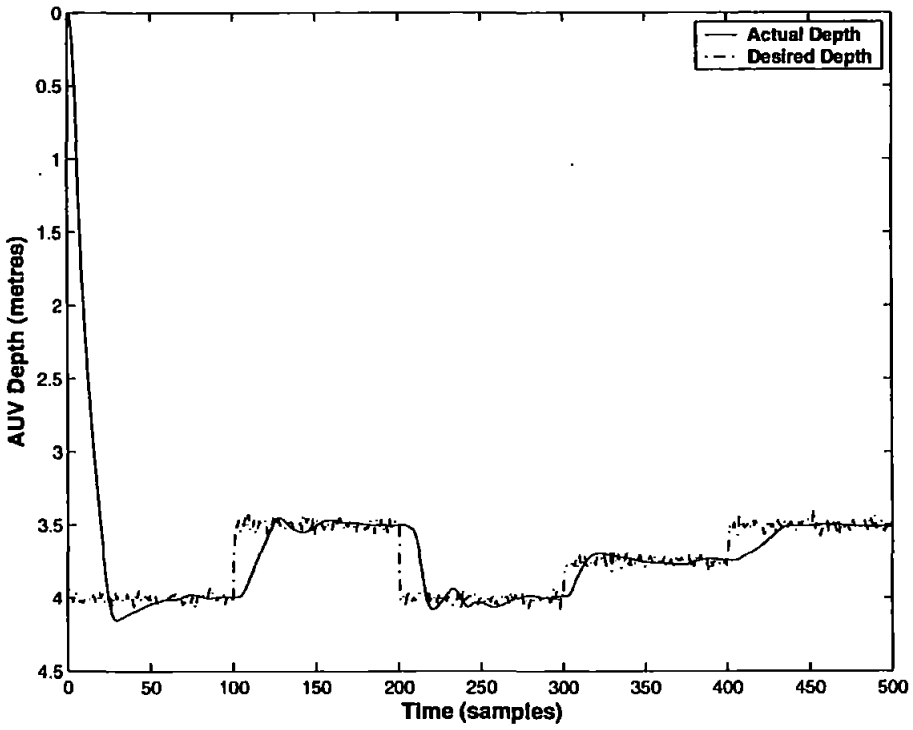


Figure 7.28: GA-MPC control of *Hammerhead* depth (altitude) where the guidance commands are generated by an onboard vision system

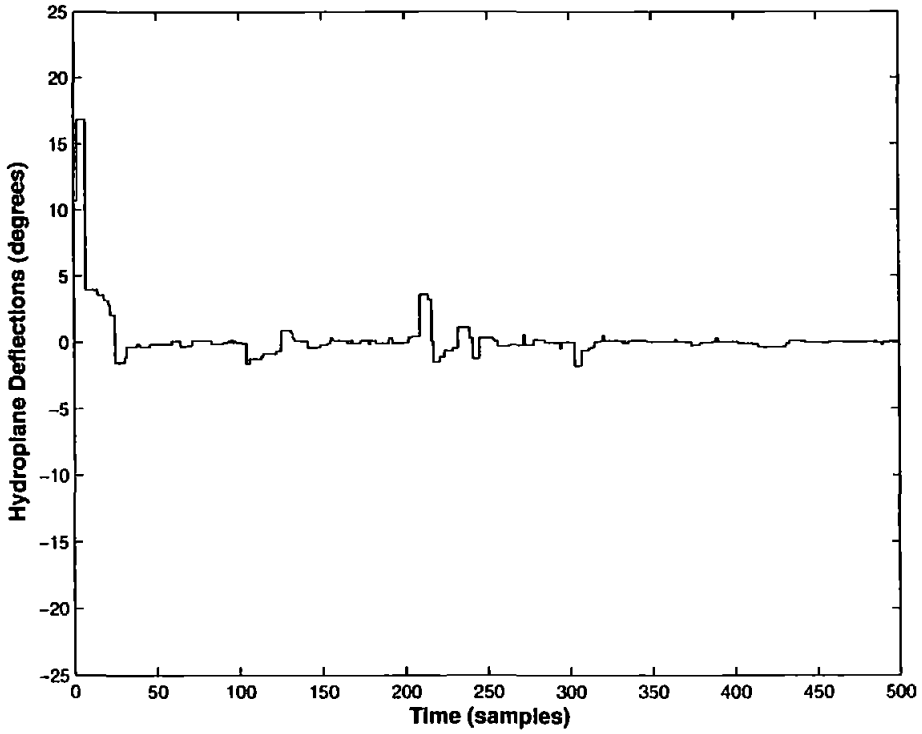


Figure 7.29: Control surface movements generated by the GA-MPC depth autopilot to track the guidance commands from an onboard vision system

At present, only linear models of *Hammerhead* vehicle are available in the horizontal and vertical planes and a further study would concentrate on developing nonlinear models of *Hammerhead* and thus able to exploit the true potential of the proposed approach. In this section, the MPC algorithm is further expanded to include nonlinear fuzzy objective functions. The fuzzy logic technique first proposed in a landmark paper by Zadeh (1965) is based on emulating human logic. The word *fuzzy* itself as is most commonly used is to describe terms which are somewhere between TRUE and NOT TRUE. The concept was later extended by Mamdani and Assilian (1975) who realised the concept of fuzzy logic control thereby developing a whole new breed of control algorithms which resembles more closely to the human decision making rather than traditional HIGH or LOW (crisp) logic. Since then, a number of applications based on Mamdani fuzzy control concept have emerged and is therefore considered to be one of the most powerful tools in contemporary control theory.

The traditional *if-then* rules, defining the fuzzy rule base to create the control logic, can be transformed into design specifications using fuzzy sets which are based on human knowledge of the problem. These fuzzy sets can be delineated to specify the objectives and constraints imposed by any system which can take the form of a cost function or performance index. The fuzzy sets are generally defined by membership functions that transform crisp values into a $[0-1]$ interval. For instance, consider the membership function shown in Figure 7.30 representing the voltage across a resistor. This membership function has *resistor voltage* as the x-axis variable whereas the y-axis represents the *degree of membership* to a given voltage. The traditional "YES NO" logic will tell if the voltage is *high* or *low* depending on a crisp boundary. However, from a human perspective, the high and low notions are not clear cut and will therefore be responded by any of the *high, not quite high, not quite low* or *low* statements. In this case, therefore, a voltage of say 3.6 volts is high but not quite high and hence deserves a membership degree of just over 0.7.

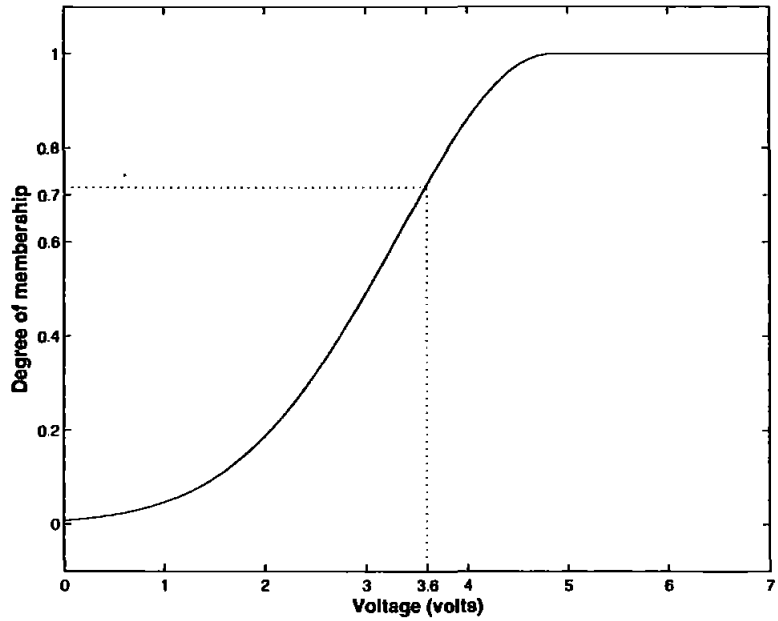


Figure 7.30: Voltage across a resistor represented as a fuzzy membership function

This fact can be exploited by using fuzzy membership functions to represent the goals and constraints of a problem at hand. For example the output error to be minimised in a given problem can be represented by an exponential membership function or other similar functions such as a bell shaped Gaussian function. Sousa and Kaymak (2001) were quick to realise this potential and employed a fuzzy cost function in an MPC framework where branch and bound (B&B) algorithm was used to solve the resulting non-convex nonlinear constrained optimization problem. The algorithm is shown to be successfully simulated on an air conditioning system which has been modelled using fuzzy techniques. Herein, the solution to the constrained nonlinear optimization problem is found using a GA (Naeem *et al.*, 2004*a*). The following section elaborates on the merits of using a fuzzy performance index followed by the choice of fuzzy membership functions and aggregation operator. Finally, the performance of the proposed control law is demonstrated on *Hammerhead* vehicle in the horizontal and vertical planes and simulations are carried out for various settings.

7.6.1 Fuzzy objective function and constraints

The fuzzy logic traditionally used as *if-then* rules can be translated to some design specifications using human expertise. These design specifications are represented in terms of an objective function which is more intuitive than the conventional cost function. There are several other reasons to choose fuzzy membership values as an

objective function to be optimized in a predictive control problem. Some of them are listed below.

Intuitiveness Fuzzy logic is derived from human expertise based on simple *if-then* statements thus making it extremely easy to comprehend.

Soft and hard constraints As opposed to traditional cost function where only soft constraint is implemented and hard constraints are dealt with separately, fuzzy objective function provides a way to realise soft and hard constraints using a single membership function which is a part of the performance index.

Normalisation The weighting matrices in a quadratic cost function are adjusted in a way so as to equalise the effect of different variables which is often performed heuristically. Employing fuzzy membership functions unravel this problem as the membership functions automatically maps the input space to a [0–1] interval.

Parameter tuning As will be shown, the weighting matrices for individual terms (membership functions) are not needed thus makes it easier to tune to any given problem.

Aggregation operator Selection of a suitable aggregation operator even reduces the tuning time further since it normally requires only a single tuning parameter for all the terms as will be illustrated later in this section.

Similar to the quadratic performance index, two variables are considered in the fuzzy objective function. The prediction error e, which is the difference between the predicted process output (in this case the vehicle heading) and reference heading, and the input variable which is the rudder angle. Separate membership functions are taken into account for the said variables which are combined together to form the objective function. For the prediction or output error, a symmetrical exponential membership function has been elected which is centred at the origin and is depicted in Figure 7.31. This means that a zero error is given a maximum membership μ_e of one while nonzero values are mapped to a positive or negative value depending on the position and magnitude of error. A nice property of exponential function is that it never decays to zero even for a significantly large error thus always providing a nonzero value.

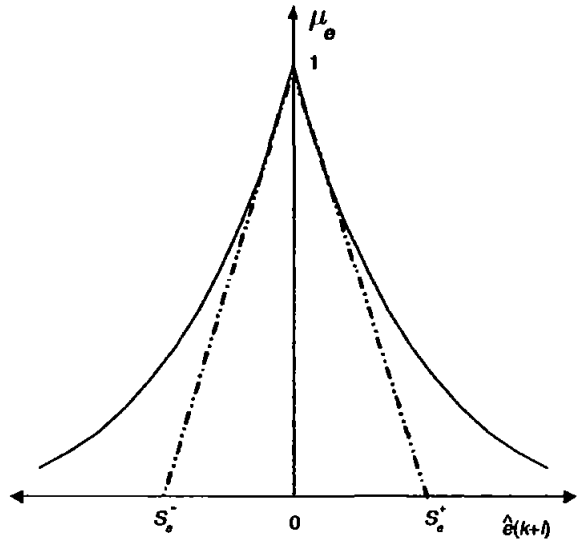


Figure 7.31: Output error membership function

Mathematically, an exponential function centred at origin is given by Equation 7.6.

$$\mu_e = \begin{cases} \exp\left(\dfrac{\hat{e}(k+i)}{S_e^-}\right), & -\infty < \hat{e}(k+i) < 0 \\ \exp\left(-\dfrac{\hat{e}(k+i)}{S_e^+}\right), & 0 \leq \hat{e}(k+i) < \infty \end{cases} \tag{7.6}$$

where $\hat{e}$ is the prediction error and $S_e^{\pm}$ represents the steepness of the exponential function. This is problem dependent and is an important tuning parameter. However, if one can identify typical error values encountered in any given problem then its quite straightforward to adjust $S_e^{\pm}$. Please note that it is not necessary for this function to be symmetrical, however, in an AUV heading control, for example, the positive and negative errors are equally probable to occur with similar magnitude scales, hence justifies the use of a symmetric function.

For the input variable, a symmetrical trapezoidal membership function which is also centred at the origin has been chosen as shown in Figure 7.32. One of the main advantages of using a trapezoidal membership function for input variable is that it automatically implements the soft and hard constraints, where u_{max} represents the maximum allowable input and $(u_{max} - u_{constraint})$ is the input bound which is allowed but not desired. Clearly, the input u is allowed to move freely within the range $(u^+_{constraint} - u^-_{constraint})$ by assigning a membership value of one. However, the

interval $(u^{\pm}_{max} - u^{\pm}_{constraint})$ is restricted by giving less weight to it thus precluding the control input selection within that region except for the satisfaction of some other criteria.

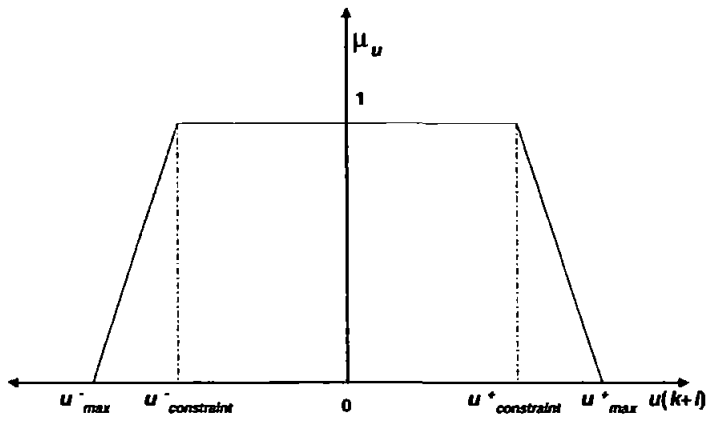


Figure 7.32: Trapezoidal membership function for input variable

Equation 7.7 provides a trapezoidal membership function in mathematical form.

$$\mu_u = \begin{cases} 1 & , \quad u^{-}_{constraint} \leq u \leq u^{+}_{constraint} \\ \dfrac{u(k+i) - u^{+}_{max}}{u^{+}_{constraint} - u^{+}_{max}} & , \quad u^{+}_{constraint} < u \leq u^{+}_{max} \\ \dfrac{u(k+i) - u^{-}_{max}}{u^{-}_{constraint} - u^{-}_{max}} & , \quad u^{-}_{constraint} < u \leq u^{-}_{max} \end{cases} \tag{7.7}$$

Again, as in the output error membership function, the trapezoidal function needs not be symmetric, however, similar arguments as in the case of error variable leads to the justification of using an even membership function.

7.6.2 Aggregation operator

Having decided upon the membership functions for available system variables, the final step is to combine all the criteria to evaluate the cost. A decision function is required which allows for interaction amongst different criteria in the objective function. A variety of aggregation operators can be chosen to be used in fuzzy predictive control such as *min* and *product t-norm*. A good account of various aggregation operators and the advantages and disadvantages of their use in predictive control has been documented (Sousa and Kaymak, 2001).

Herein, the Yager t-norm has been chosen as the decision function since it uses only one parameter to tune the objective function and hence interact amongst different criteria. Moreover, this operator covers the entire range of t-norms, i.e., it goes from the drastic intersection to the minimum operator (Sousa and Kaymak, 2001). The fuzzy cost function and Yager t-norm are given by Equations 7.8 and 7.9 respectively.

$$\bar{\mu}_c = \sum_{i=1}^{H_p} (\bar{\mu}_e(\hat{e}(k+i)))^{w_Y} + \sum_{j=1}^{H_c} (\bar{\mu}_u(u(k+i)))^{w_Y} \tag{7.8}$$

$$\mu = \max(0, 1 - \bar{\mu}_c^{1/w_Y}) \tag{7.9}$$

where $\bar{\mu} = 1 - \mu$ and $w_Y > 0$ is the tuning parameter. Please note that there is only one tuning variable involved in the objective function in contrast to the traditional cost function where the number of parameters to be adjusted depend on the number of variables in the cost function.

7.7 Simulation Results

The control algorithm is now tested in simulations and the performance is evaluated based on the guidance laws developed in Chapter 5 and integrated with other controllers. It is shown that the control system is capable of producing fine results by incorporating knowledge of human expertise in the form of an objective function. The implementation methodology of hard constraints is also shown to be quite effective in that the controller never violates the actuator constraints. Several results are illustrated to assess the heading and depth autopilots performance. An altitude controller is also formulated to maintain the demanded altitude where the guidance commands are assumed to be issued by an onboard vision system.

7.7.1 Heading control

The proposed control algorithm is simulated in this section for several situations. A fair comparison of the control scheme using fuzzy decision function with the one minimising a quadratic performance index would require the response to be evaluated for the same scenarios. A simple heading step response of the closed loop system is evaluated to start off followed by waypoint tracking with and without considering sea currents. The hybrid and PNG guidance schemes are then integrated to demonstrate the tracking capabilities of the proposed controller. In this case, the weighting ma-

trices Q and R in the quadratic objective function are replaced by the parameters S_e and w_Y in a fuzzy performance index. The hard constraints on the rudder are moved to the full $\pm 22°$ therefore u^-_{max} and u^+_{max} in Figure 7.32 are taken as $-22°$ and $+22°$ respectively. However, the control is only allowed to move freely within the range $\pm 20°$ to avoid saturation and hence any nonlinear behaviour. Therefore, $u^{\pm}_{constraint}$ in Figure 7.32 is set equal to $\pm 20°$.

Simulations are carried out first for a step change in heading. The vehicle is launched with an arbitrary orientation and is required to follow a specified heading. The parameters S_e and w_Y are chosen as 0.5 and 2 respectively whereas the GA parameters are provided in Table 7.6 and are selected to minimise the control effort and increase the speed of response. It should be noted that the parameter S_e is selected based on

Controller Parameters	Step response
H_p	20
H_c	1
Mutation prob.	0.005
Crossover prob.	0.1
No. of generations	10
Population size	100
Insertion rate	0.5

Table 7.6: GA-MPC tuning parameters using a fuzzy decision function for heading control (step response)

the measurement of expected error in the output response, which is the AUV heading in this case. Consider the worst case scenario of vehicle pointing somewhere close to 0°, whereas the desired heading is around 360° which is tantamount to 2π radians. This gives a maximum initial error of approximately 6.28 radians $(2 * \pi)$ moving in a clockwise direction. Therefore, a selection of S_e beyond this value is not justified since this initial error value is always going to decrease no matter in which direction the vehicle manoeuvres. Additionally, since the GA is designed to minimise whereas the fuzzy performance index represents the membership value of the given variables which is to be maximised. Therefore, this needs to be transformed into a minimisation problem for the GA to optimize it properly. This could be done in several ways such as changing the sign of the cost function or taking its reciprocal. Herein, the reciprocal technique is used given by Equation 7.10, since it has an additional advantage of mapping the objective function values into a $[0 - 1]$ space.

$$J = \frac{1}{1+\mu} \tag{7.10}$$

The step response of the closed loop system is depicted in Figure 7.33 showing that the vehicle is closely following the set point with little overshoot and zero steady state error. The response is quite similar to the one in Figure 7.6 obtained by the minimisation of a quadratic performance index. In fact the settling time for both controllers is approximately the same except for the small overshoot present in the former case. The canard demand is also shown in Figure 7.34 requiring minimum control effort and stays within the specified constraint of $\pm 22°$. However, the bulk of the rudder movement is restricted to $\pm 20°$ as imposed by the $u^{\pm}_{constraint}$ variable in the input membership function.

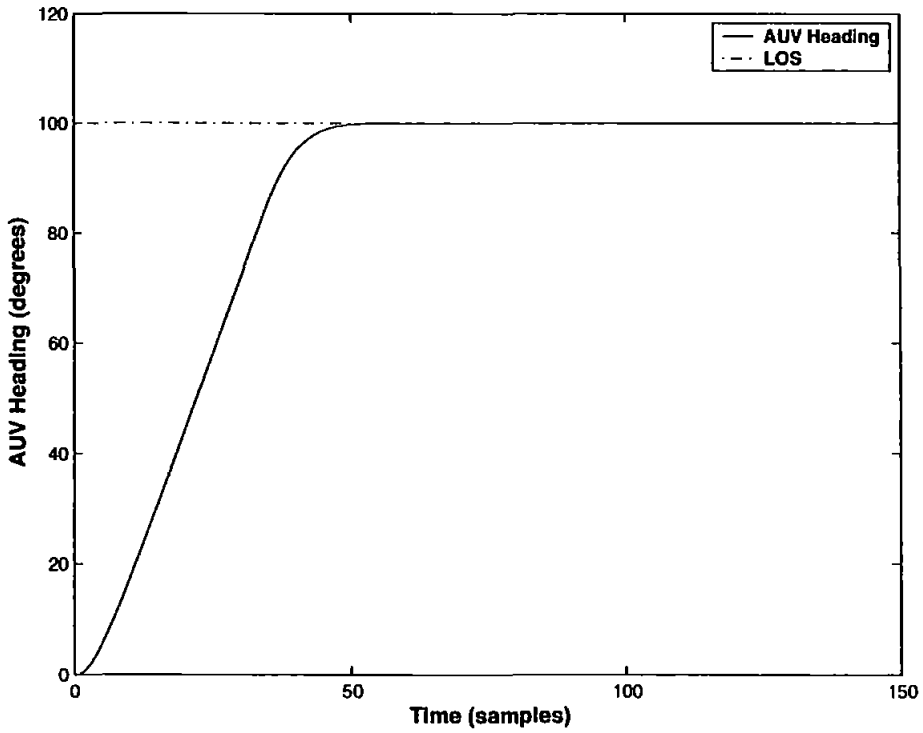


Figure 7.33: Step change in heading response of the GA-MPC controller with fuzzy objective function

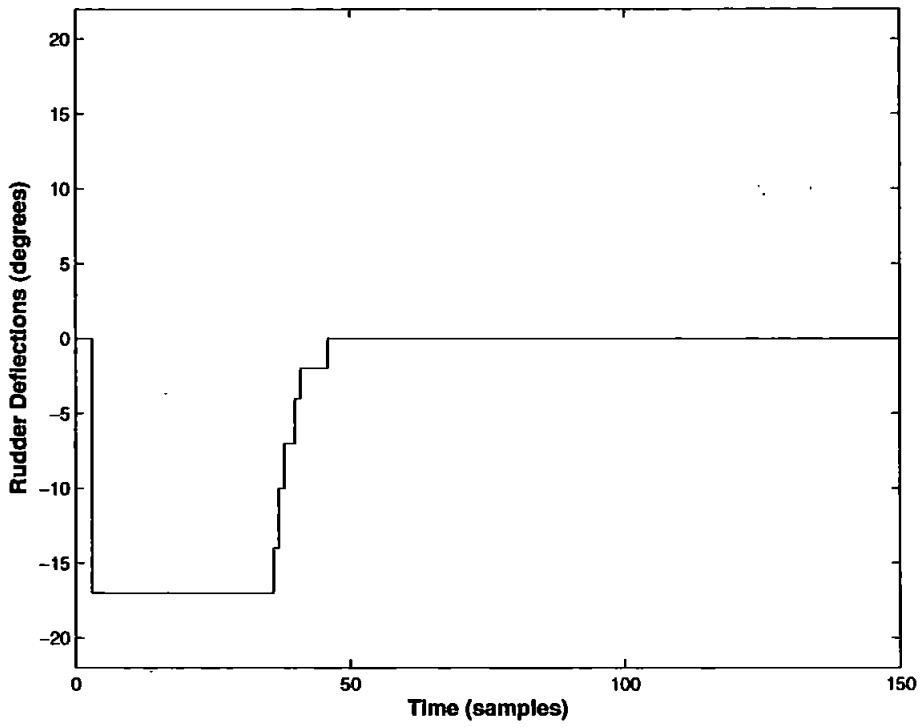


Figure 7.34: Optimal rudder deflections generated by the GA-MPC controller with fuzzy objective function

Waypoint following

Next, the control law is simulated for waypoint following where the intent is to track all the specified waypoints stored in the vehicle mission planner. The affect of sea currents on vehicle's trajectory is also investigated and compared with the AUV's course when quadratic cost function was used as the performance index. Simulation is carried out first for waypoint following without any currents and is subsequently modified to include the effect of disturbances. The initial AUV coordinates in the two-dimensional frame of reference are $(0, 10)$ whilst the stored waypoints are as given in Table 6.1. The next waypoint is selected when the vehicle enters a COA around the waypoint of radius $10m$. The GA parameters for waypoint following are provided in Table 7.7 whereas S_e and w_Y are selected as 1.5 and 1.8 respectively.

Controller Parameters	Waypoint following
H_p	20
H_c	1
Mutation prob.	0.008
Crossover prob.	0.1
No. of generations	3
Population size	250
Insertion rate	0.1

Table 7.7: GA-MPC tuning parameters using a fuzzy decision function for heading control (waypoint following)

The resulting AUV trajectory is illustrated in Figure 7.35 showing in addition, the waypoints and the trajectory without any dynamics.

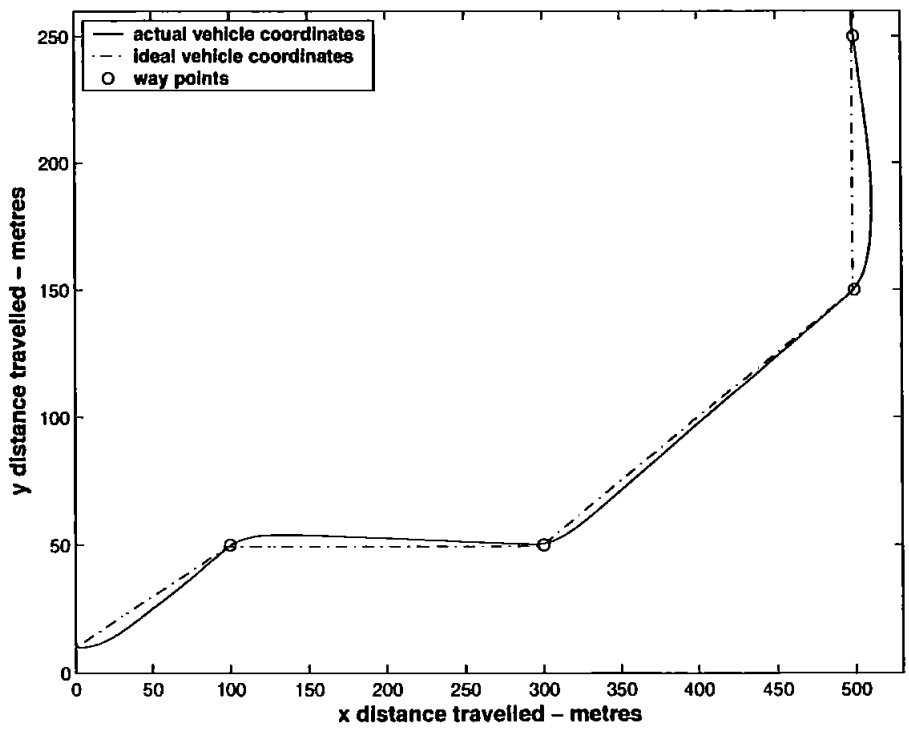


Figure 7.35: AUV trajectory and target position coordinates in a way point tracking mission without any disturbances using a GA-MPC controller with a fuzzy cost function

As shown, the response is slightly sluggish as compared to the former case where

the ideal and AUV trajectories almost super impose on each other. However, the vehicle successfully passes through all the waypoints. The control surface deflections are also depicted in Figure 7.36 where the maximum deflection observed is less than the previous case which explains the sluggishness of the response. This is because the input variable is heavily penalised which restricts its movement within the constraints. Further tuning of the w_Y variable in the objective function could help reduce this discrepancy.

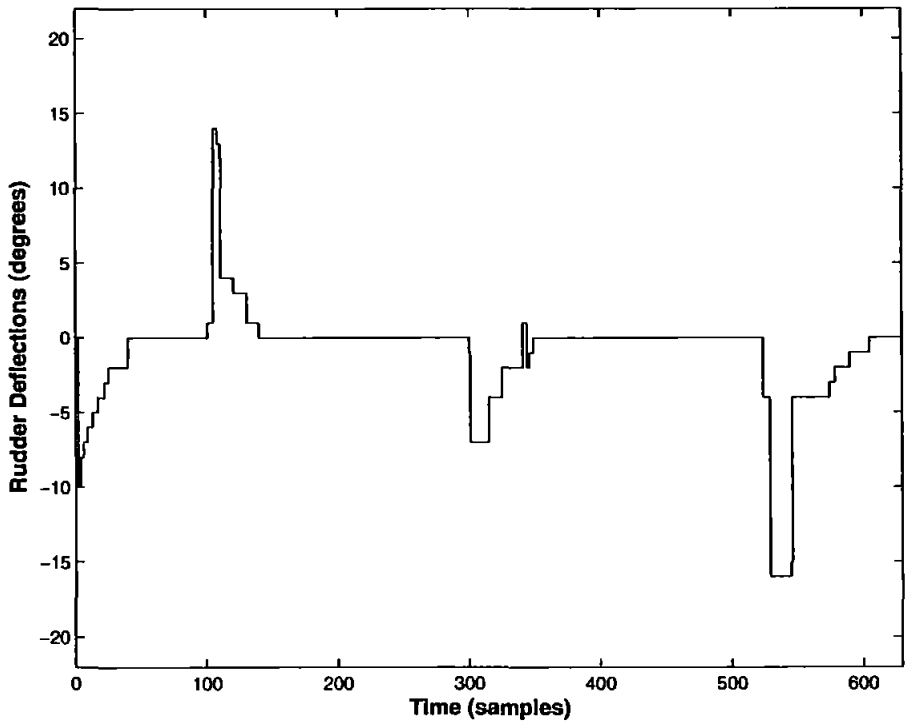


Figure 7.36: Rudder deflections generated by the controller needed to track the waypoints without any disturbance using a GA-MPC controller with a fuzzy cost function

The vehicle heading angle shown in Figure 7.37 is also closely following the desired LOS angles between any two successive way points. The settling time is distinctly higher than the previous case due to above mentioned reasons. Please note that the MSE of the vehicle's trajectory is approximately $9m^2$ which is quite high as compared the former case. This disagreement could be explained by estimating the standard deviation of rudder deflections which turns out to be 3.5^0. Clearly, the control effort in this case is less by 1.5^0 that made the response slightly slower but still adequate for testing purposes.

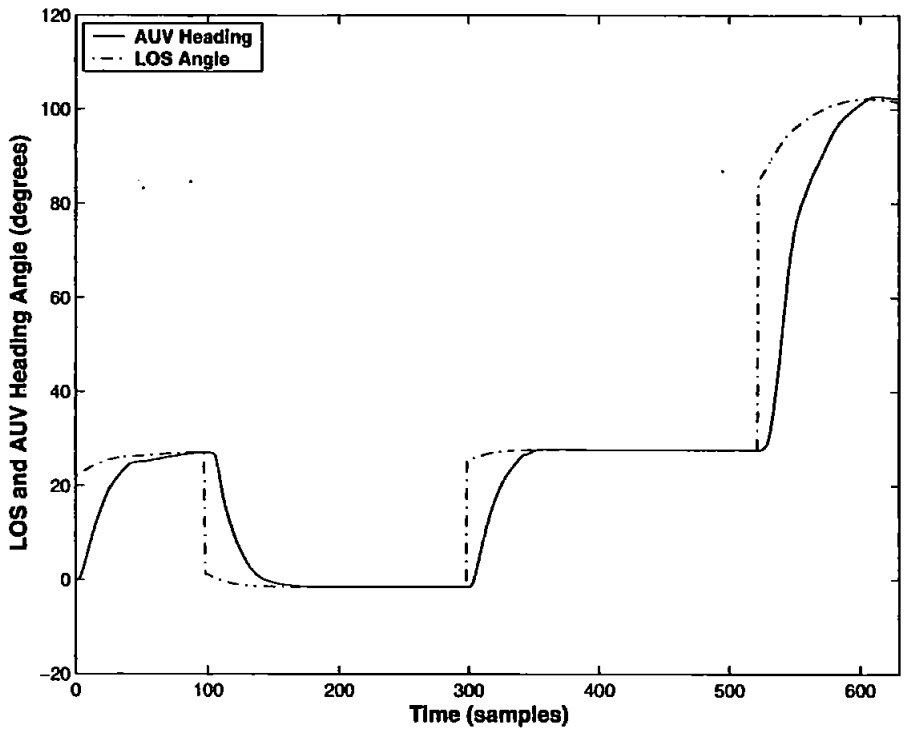


Figure 7.37: Desired yaw and vehicle heading angles in case of a fuzzy cost function in a waypoint mission scenario without any disturbance

Finally, a sea current is assumed to be acting on the vehicle in the positive y-direction having a magnitude of $0.5m/s$. The resulting closed loop performance is illustrated in Figure 7.38 showing the affects of sea currents on vehicle's trajectory. The disturbance is striving to push the AUV off the track in the direction of current, however, the controller keeps the vehicle on the desired course. The control effort is depicted in Figure 7.39 showing vigorous rudder movements in response to the change in vehicle's heading due to sea currents, however, it never violates the imposed constraints. The desired heading or LOS angle between actual AUV position and the waypoints is plotted in Figure 7.40 which is varying continuously because of the addition of disturbances, however, the vehicle is closely tracking the required heading angles. A MSE of $1390m^2$ is obtained which is again much higher than the previous case when the disturbance was included. The standard deviation of rudder deflections, however, is approximately the same.

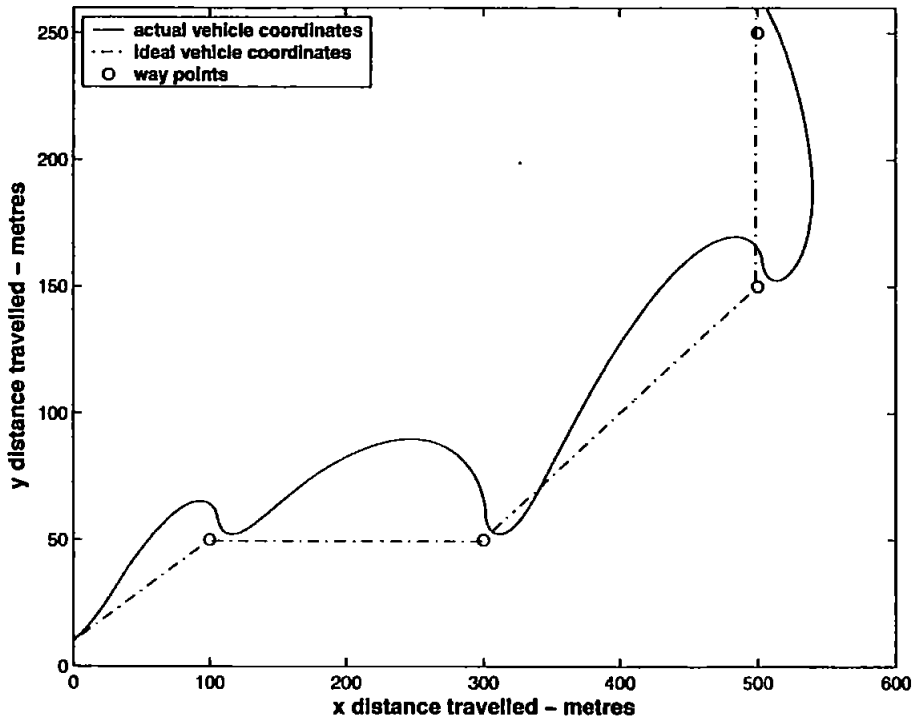


Figure 7.38: AUV trajectory and target position coordinates in case of a fuzzy cost function in waypoint following with a sea current of $0.5m/s$ in the positive y-direction

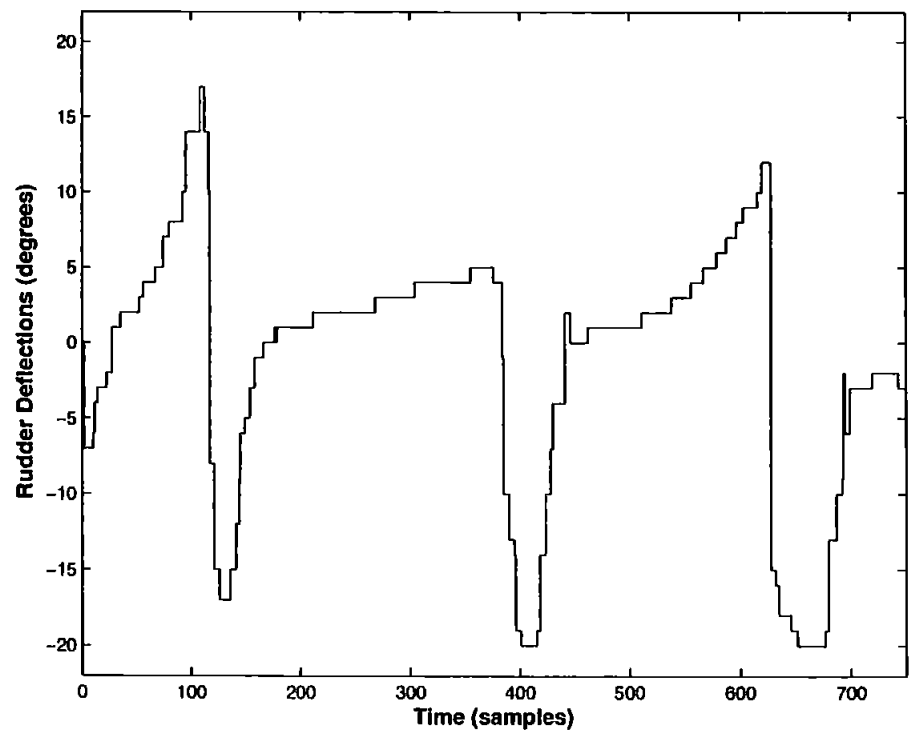


Figure 7.39: Rudder deflections generated by the controller with a fuzzy cost function needed to track the waypoints with sea current disturbance in the positive y-direction

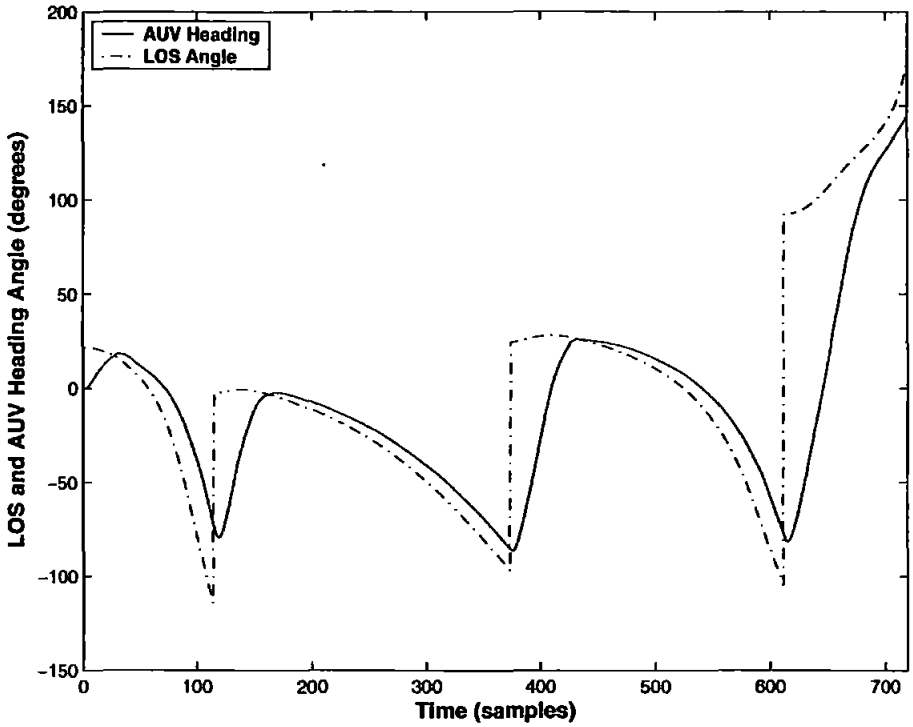


Figure 7.40: Desired yaw and AUV heading angle in a waypoint following mission showing the affects of sea current disturbance

Hybrid guidance law

The proposed hybrid guidance law presented in Chapter 5 and simulated with an LQG/LTR autopilot and GA-MPC using quadratic performance index is finally tested with a GA-MPC employing a fuzzy objective function. Controllers are developed for each speed model all of whose parameters turned out to be the same for optimal performance. Table 7.8 summarises the GA variables whilst the fuzzy objective function parameters S_e and w_Y are selected as 1.5 and 1.8 respectively. The result of the integrated hybrid guidance and control simulation is depicted in Figure 7.41. The cable tracking response of the controller resembles with the GA-MPC output when conventional cost function was employed with a smooth output for varying speeds and overshoot and oscillations at a fix velocity. However, as in the former case, the controller clearly outperforms the LQG/LTR autopilot controller which provides a poor response during turning manoeuvres at fix speed.

The heading plot shown in Figure 7.42 bears similar characteristics as in Figure 7.21 for the case of quadratic objective function. The LOS is constant at approximately

Controller Parameters	Cable tracking using hybrid guidance
H_p	15
H_c	1
Mutation prob.	0.005
Crossover prob.	0.1
No. of generations	10
Population size	100
Insertion rate	0.4

Table 7.8: GA-MPC controller parameters to minimise the fuzzy objective function and integrated with a hybrid guidance strategy

-29^0 during the launching and midcourse phase. By reducing the vehicle velocity in the terminal and tracking phases, the oscillations are avoided in the heading output. Finally, the control surface deflections in Figures 7.43 and 7.44 for both variable and fix speed cases shows slightly higher movements at lower velocity whereas the magnitude is quite small at a fix velocity.

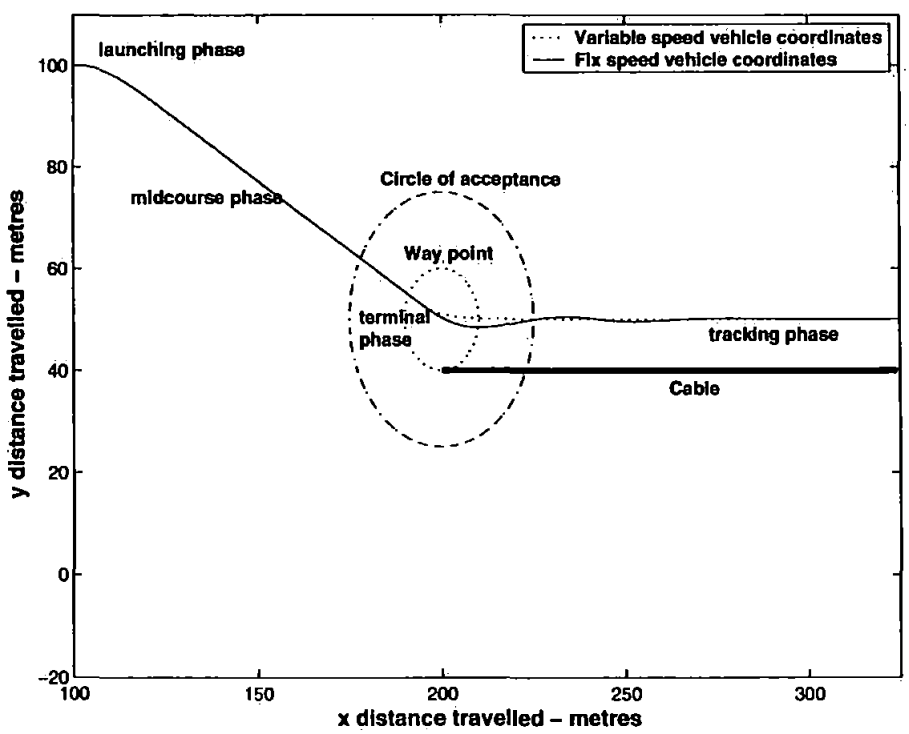


Figure 7.41: AUV trajectory at fix and variable velocities in a hybrid guidance strategy integrated with a GA-MPC controller with a fuzzy cost function

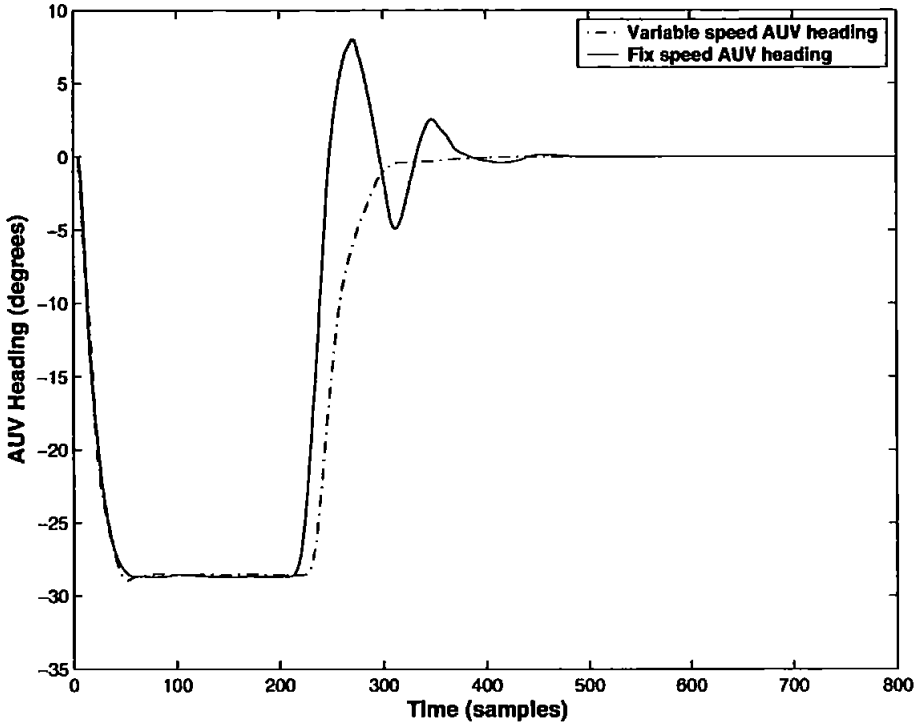


Figure 7.42: AUV heading at fix and variable velocities in a hybrid guidance strategy integrated with a GA-MPC controller with a fuzzy cost function

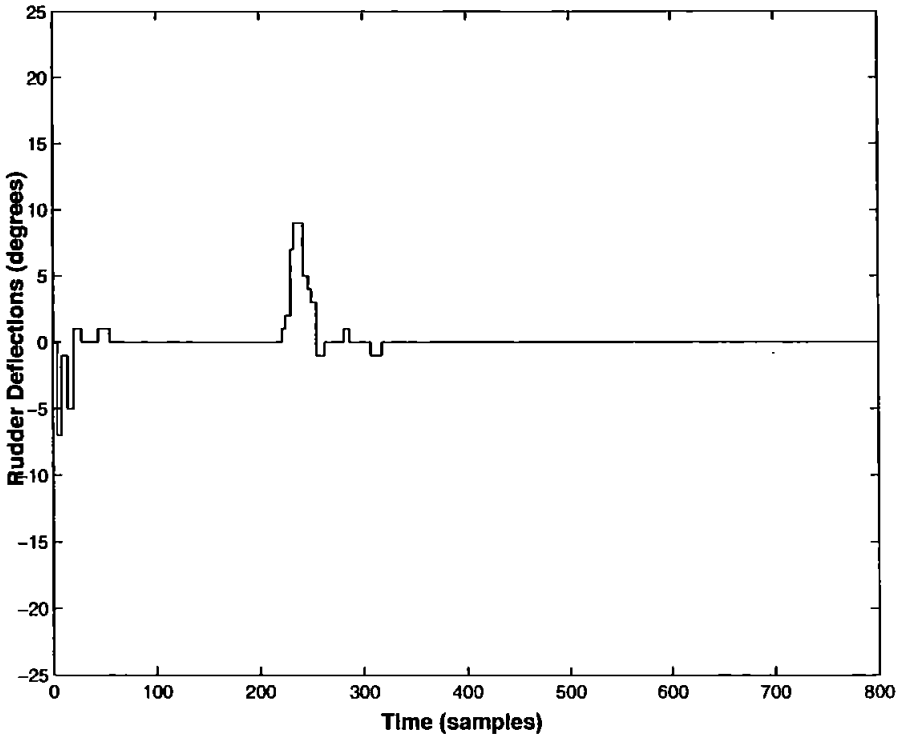


Figure 7.43: Rudder deflections at variable velocities in a hybrid guidance strategy integrated with a GA-MPC controller with a fuzzy cost function

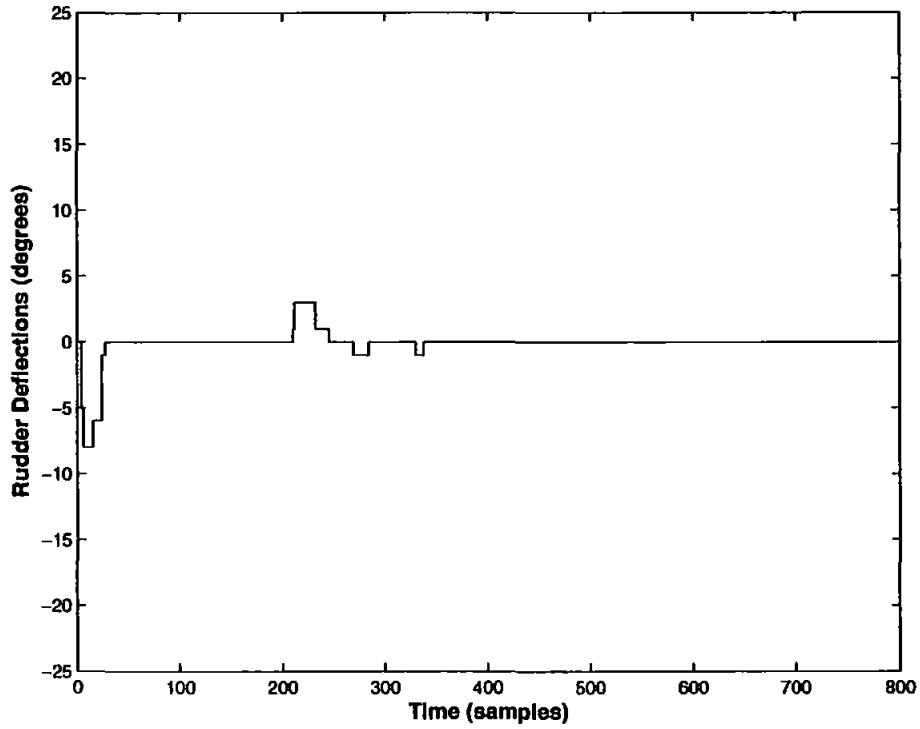


Figure 7.44: Rudder deflections at a fix velocity in a hybrid guidance strategy integrated with a GA-MPC controller with a fuzzy cost function

Proportional navigation guidance

A GA-MPC heading controller using fuzzy objective function is finally integrated with the PNG system developed in Chapter 5. Simulation results are presented here and compared with the other controllers developed antecedently. A description of the proposed integrated system has been outlined before. The controller parameters are provided in Table 7.9 whereas S_c and w_Y are chosen as 0.5 and 2 respectively.

The performance of the guidance and control system is depicted in Figure 7.45 showing excellent cable tracking characteristics. The response, however, is similar to the previous cases and no further improvement is observed. The heading angle is also plotted in Figure 7.46 including the PNG guidance commands. The vehicle's initial heading is arbitrary and assumed to be 0^0 whilst the first guidance command issued is -90^0. The control surface deflections shown in Figure 7.47 are also within the constrained limits of $\pm 20^0$ and constitute minimum control energy expense.

Controller Parameters	Cable tracking using PNG guidance
H_p	10
H_c	1
Mutation prob.	0.005
Crossover prob.	0.7
No. of generations	20
Population size	100
Insertion rate	0.5

Table 7.9: GA-MPC tuning parameters using a fuzzy decision function and integrated with a PNG guidance system

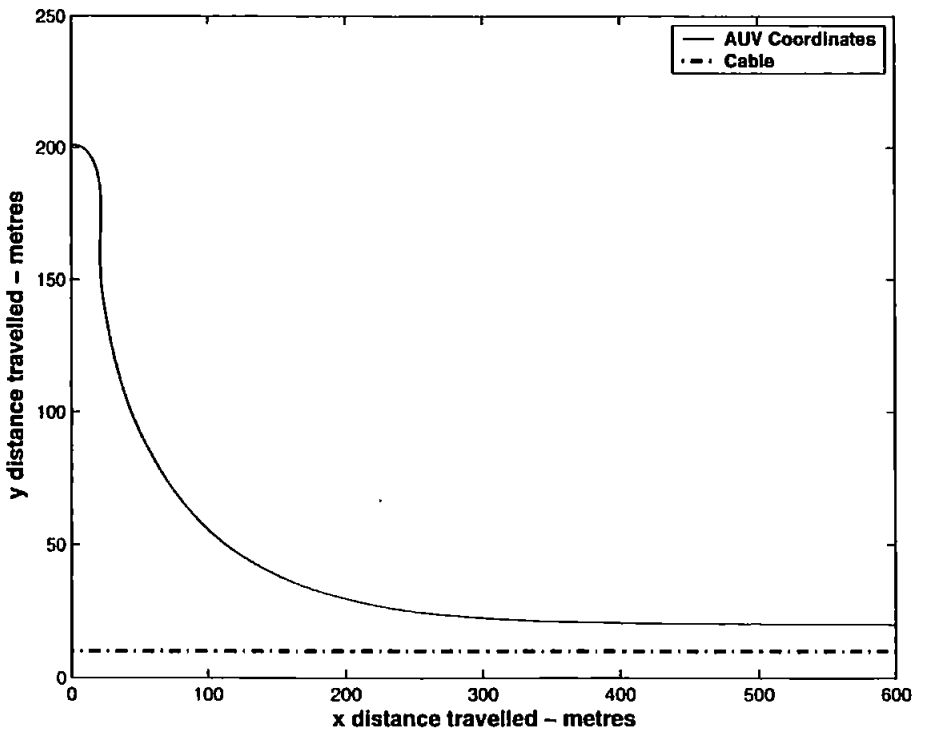


Figure 7.45: AUV trajectory showing excellent cable tracking response using a PNG strategy integrated with a GA-MPC with a fuzzy cost function

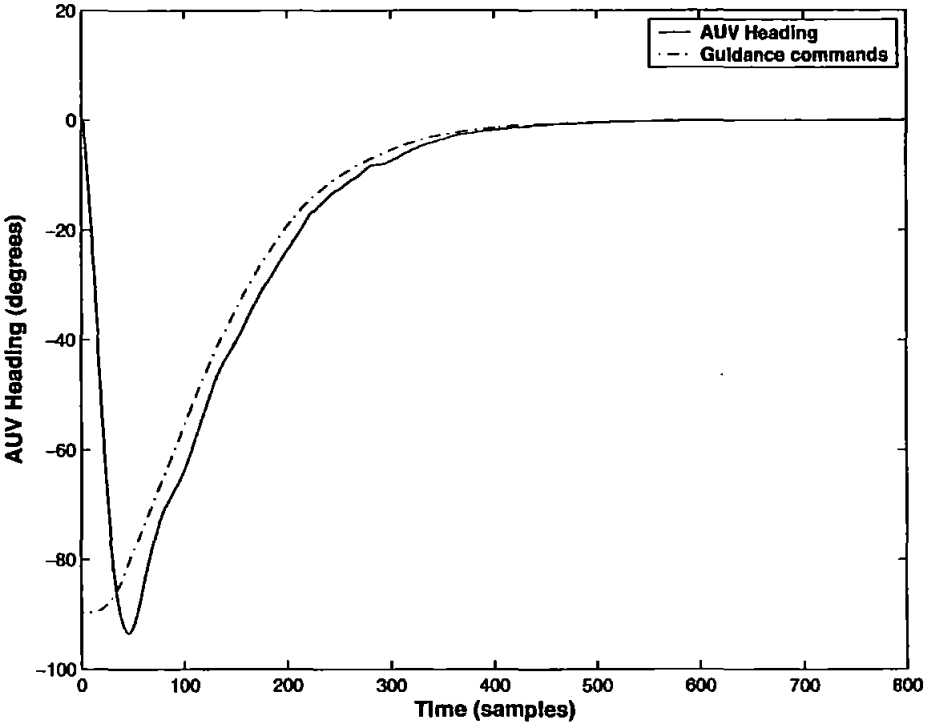


Figure 7.46: *Hammerhead* heading response closely following the guidance commands in a PNG strategy integrated with a GA-MPC controller with a fuzzy cost function

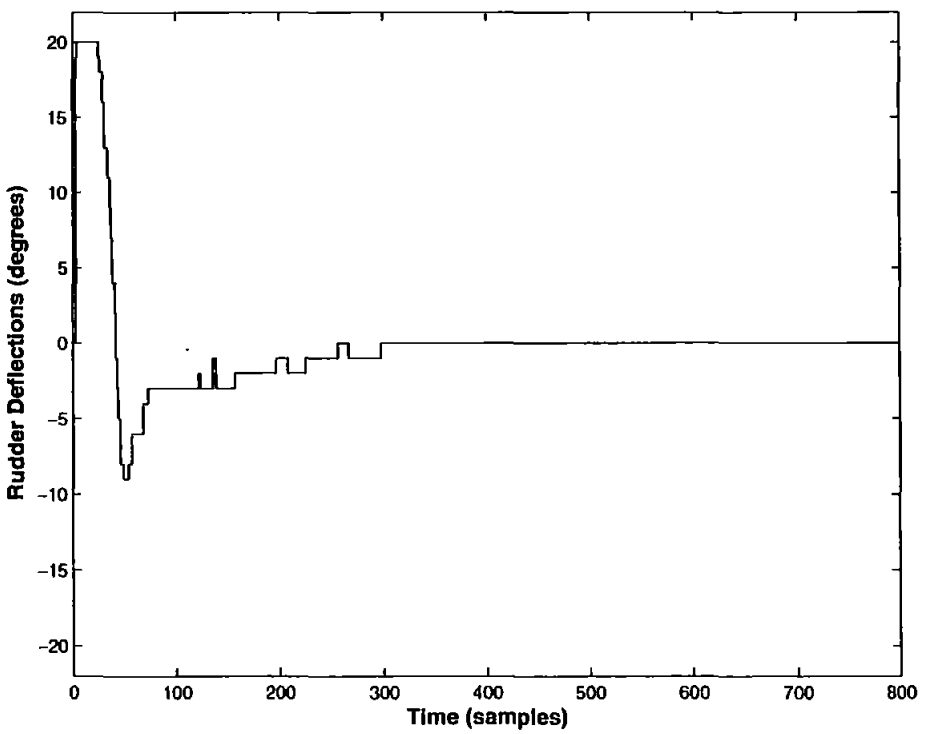


Figure 7.47: Rudder deflections needed to track the cable in a PNG strategy integrated with a GA-MPC controller with a fuzzy cost function

7.7.2 Depth control

Lastly, the GA-MPC algorithm using a fuzzy objective function is simulated for *Hammerhead* depth control. The depth dynamic model in Equation 4.23 serves the purpose of simulating the depth response to validate the proposed control law. A step change in depth is demanded when the vehicle is on the surface. The input membership function variables are selected as $\pm 25^{\circ}$ for $u^{\pm}_{max}$ and $\pm 22^{\circ}$ for $u^{\pm}_{constraint}$. S_e and w_Y are both adjusted to 1.2 whereas the other controller parameters are given in Table 7.10.

Controller Parameters	Depth Step response
H_p	20
H_c	1
Mutation prob.	0.005
Crossover prob.	0.1
No. of generations	10
Population size	100
Insertion rate	0.5

Table 7.10: GA-MPC tuning parameters using a fuzzy decision function for depth control

The *Hammerhead* depth response to a step input of $3m$ is depicted in Figure 7.48. Clearly, the vehicle approaches the desired depth steadily without any overshoot and no steady state error. In this case, the settling time is slightly higher than the previous case because of the weighting on control input shown in Figure 7.49. The maximum deflection observed here is approximately 7^0 while it was around 10^0 in the previous case. Adjusting the tuning parameter w_Y can reduce this disagreement or even improve the performance of the control algorithm.

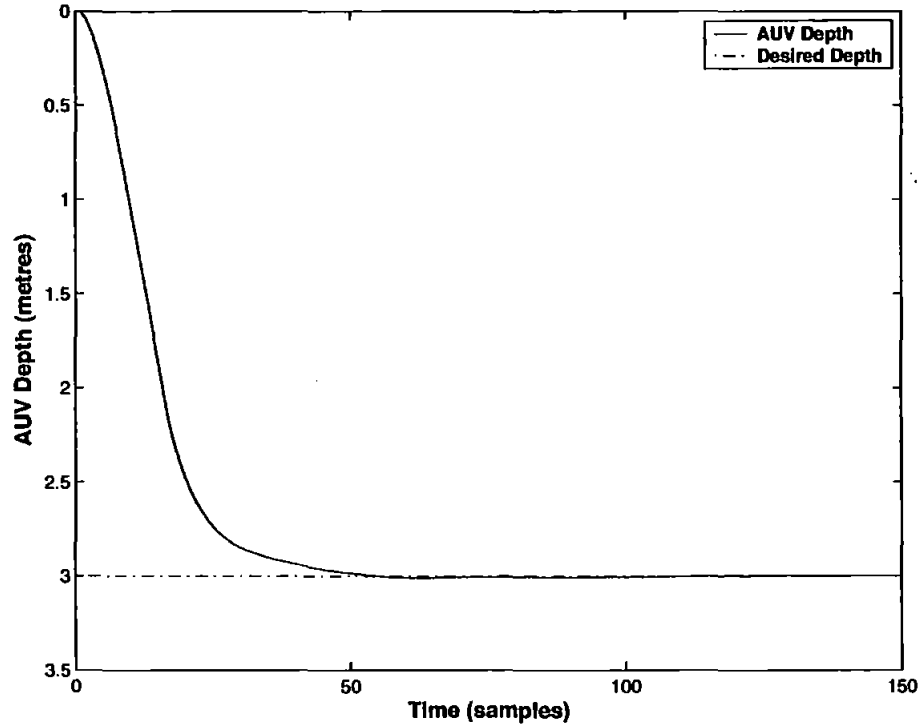


Figure 7.48: Depth response of *Hammerhead* to a step input using a GA-MPC controller with fuzzy objective function

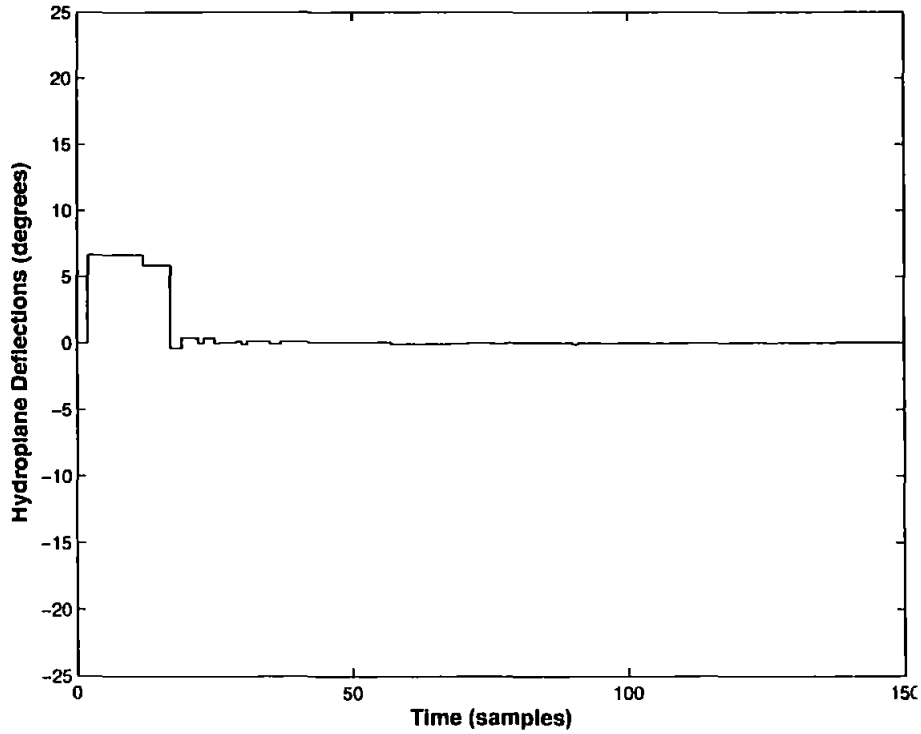


Figure 7.49: Control surface deflections generated by the GA-MPC controller for a depth control mission using fuzzy objective function

Altitude control

The mission scenario of altitude control has been explained before. Herein, simulation results are illustrated and the performance is compared with the LQG/LTR controller and GA-MPC autopilot employing a quadratic cost function. The same depth controller is used here as developed in the preceding subsection. The depth controller is assumed to be integrated with an onboard vision system which provide guidance commands to be followed. The guidance system generates altitude information which is converted into desired depth demands and hence followed by the depth autopilot. A normally distributed noise is also added to simulate the uncertainty in the guidance input. The plot in Figure 7.50 evidently shows the noisy guidance input and vehicle heading. The response show excellent disturbance rejection characteristics of the controller as compared to the LQG/LTR output which tracks all the noisy input commands and constitute a tremendous waste of control energy. The heading response is slightly slower than the conventional cost function case due to weight on control input. Figure 7.51 depict the control surface movements needed to track the desired depth which are well within the constrained limits with minimum control energy expense.

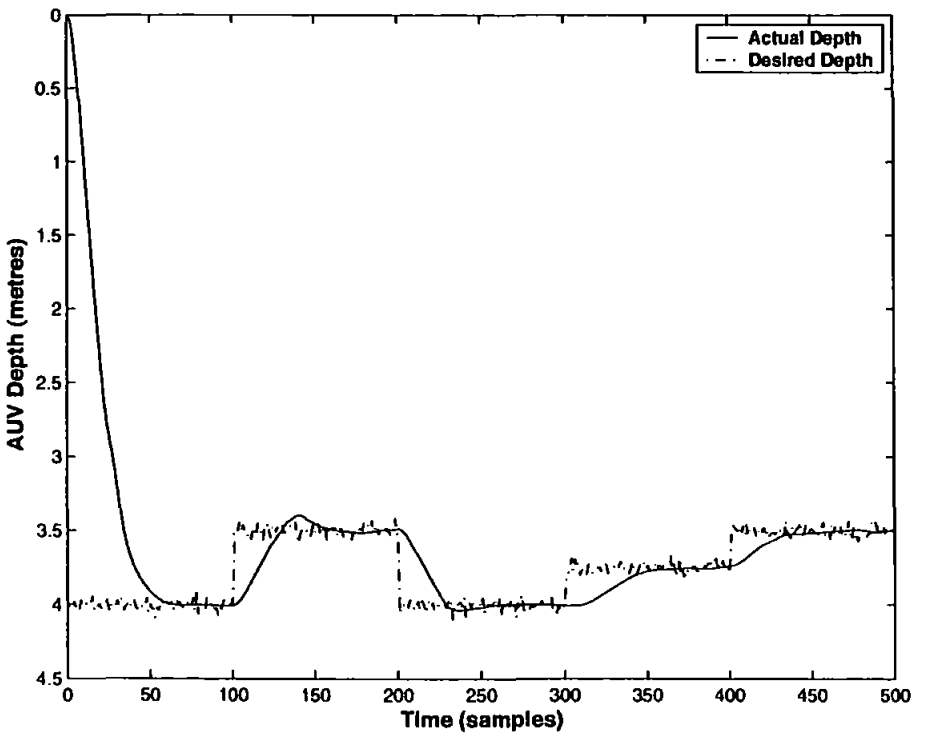


Figure 7.50: GA-MPC autopilot with fuzzy cost function integrated with an onboard vision system showing that the vehicle is closely following the desired depth (altitude)

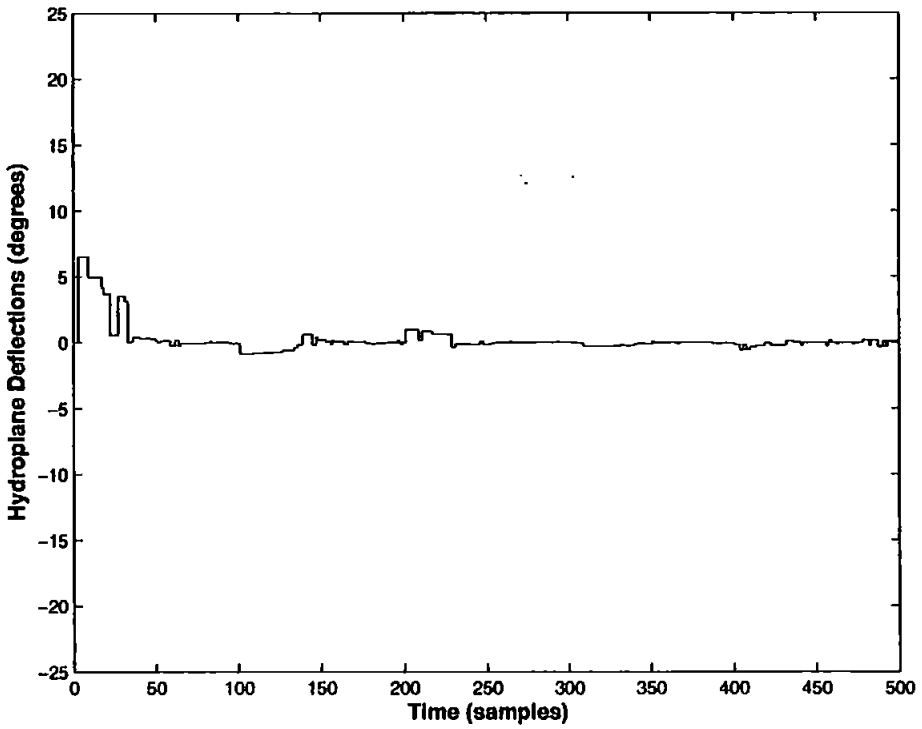


Figure 7.51: Hydroplane deflections generated by the GA-MPC autopilot with fuzzy cost function to track the desired depth (altitude) demands generated by an onboard vision system

7.8 Concluding Remarks

This chapter introduced a novel and robust control methodology for underwater vehicle control. A model based predictive controller is considered which has gained a widespread interest in numerous controller applications owing to its success in the process industry. Two modifications to the standard MPC problem are suggested to be implemented in the *Hammerhead* AUV and simulation results were illustrated in the horizontal and dive planes to exemplify the efficacy of the proposed techniques. It should be stressed here that the presented approaches are fairly original to the underwater research field. The MPC has been quite successful in the process industry however, the application in other areas is still limited. It is one of the aims here to introduce another robust control methodology to the field of underwater robotics. The models used for simulation are the ones extracted from test trials data using SI in Chapter 4. In addition, for the hybrid guidance law simulation, an AUV model has been borrowed from literature which is presented in terms of vehicle velocity.

The first approach unveiled was the application of GA to optimize the traditional quadratic cost function in a predictive control frame work. The algorithm was simulated for different settings such as tracking LOS angles, waypoint following and cable tracking missions. In the other proposed strategy, the standard error cost function in the MPC is replaced by a fuzzy performance index which represents the goals and constraints of the problem and a GA is employed for optimization. Simulation results are also shown in this case for set point tracking in both planes including way point following with and without taking into account the sea current disturbances. The performances of the proposed control schemes were assessed and a qualitative and quantitative comparison is made with an LQG/LTR autopilot developed in Chapter 6 which is summarised in Table 7.11 below.

	LQG/LTR Controller		GA-MPC Controller			
			Conventional objective function		Fuzzy objective function	
	MSE (m^2)	STD (°)	MSE (m^2)	STD (°)	MSE (m^2)	STD (°)
without sea currents	9.6	6	1.96	5	9	3.5
with sea currents	698	9	1049	9	1390	9

Table 7.11: Quantitative comparison of the *Hammerhead* autopilots developed in Chapters 6 and 7

The table shows the STD of the rudder deflections and the MSE between ideal and actual AUV trajectories with and without the presence of disturbances. The GA-MPC with the quadratic objective function appears to be the best candidate in the absence of sea currents with minimum MSE and STD. The MSE of the remaining controllers is however the same with a higher STD for the LQG/LTR autopilot. A similar analysis when the effects of sea currents was included reveals that with the same STD of rudder deflections, the LQG/LTR controller outperforms the other autopilots in minimum MSE sense for this scenario only. However, the overall performance of the GA-MPC with or without standard cost function is much better than LQG/LTR controller in terms of generating optimum rudder deflections. This is confirmed by comparing Figure 6.30 with Figure 7.20 for the hybrid guidance simulation, where the LQG/LTR performance is worse at fixed vehicle velocity. The GA-MPC on the other hand is far superior than LQG/LTR at both fixed and variable speeds. In addition, for the pure pursuit guidance scenario, although the tracking performances are similar, there is a

peak rudder deflection from positive to negative side in Figure 6.25 which is absent in Figure 7.13 and very small in Figure 7.47. The results produced by both the GA-MPC autopilots in all situations are much more realistic from an implementation perspective.

The next chapter concentrates on the application of the proposed controllers in real time in the *Hammerhead* vehicle. Problems encountered during the experiments are described and discrepancies in simulation and real time results are elucidated.

Chapter 8

Experimentation with the *Hammerhead* AUV

The ultimate step in a controller design is to evaluate its performance in a real time environment. This final chapter on control system design deals with experimenting the autopilot developed in the previous chapter on the *Hammerhead* AUV. The controllers were tuned using a model of the actual physical system and will now be tested to gauge their capabilities. Results are shown here for the GA-MPC autopilot only. This is because the trials could not be conducted for the other controllers due to the lack of time and therefore will not be included in this thesis. It is demonstrated through results that the controller is quite robust under different operating conditions and modelling uncertainty. To the author's knowledge, this is the first successful application of a GA in real time optimization for controller tuning in the marine sector and thus this thesis makes an extremely novel and useful contribution to control system design in general.

8.1 Introduction

The primary aim of developing the *Hammerhead* AUV is to provide an easy to use test platform for other underwater research groups within the UK to test their NGC algorithms. Hence, the vehicle command and control system must be flexible enough to accommodate various requirements set by the user. In this penultimate chapter, the current hardware and software setup is explored. Flow charts depicting the mission sequence are presented involving communication between onboard sensors, actuators and the control computer. The results of the autopilot implementation are then presented which has been developed previously.

There is a distinct lack of experimental results sensed in the available material for underwater research where the majority of work is based around controller testing in simulations with very little or no emphasis on its real-time counterpart. It is also the aim of this thesis to provide real-time results based on contemporary control theory and thus makes an invaluable contribution to the underwater research literature. Moreover, it should be stressed here that the autopilots which are to be implemented in the *Hammerhead* vehicle are based on advanced control concepts which to the author's knowledge have not yet been exhaustively tested in the marine sector. For an interested reader, Lea (1998) reviewed various control systems for AUVs presenting experimental comparative tests, whilst Craven *et al.* (1998) report their findings on classical and advance controllers for AUVs.

It has been observed that for practical applications, the majority of AUVs around the world employ simple control algorithms such as the proportional, integral and derivative (PID) controller. Clearly, this strategy is useful in commercialising the product in a short time by employing not so easy to tune controller which can be realised in hardware in a straightforward manner. For instance, the Natural Environmental Research Council's *AUTOSUB* vehicle based at the Southampton Oceanography Centre, UK, uses PD controllers to control the vehicle position, depth and altitude which is shown to be adequate (McPhail and Pebody, 1998). The underlying philosophy for selecting the PD controller is "keep it simple" for the delivery of a reliable and maintainable system. However, it is shown that the controller could not cope with changes in water density (resulting from a combination of fresh and saline water), which results in the loss of vehicle buoyancy causing it to sink (Millard *et al.*, 1998). Similary, the *AQUA Explorer* (Asakawa *et al.*, 1996) developed by KDD R&D Laboratories, Japan, and the *Twin Burger* (Balasuriya and Ura, 1998) vehicle based at the University of Tokyo, employed PID and PD control algorithms for attitude control. This does not prevent researchers from developing a new breed of control laws suitable for underwater environment which are robust and overcome many difficulties inherent in PID controllers. For example, in addition to the PID, a fuzzy control law is developed for the *AQUA Explorer* and experimental results for the case of a cable tracking mission are reported (Ito *et al.*, 1994*b*; Kato *et al.*, 1994*b*). A neural network based control system has also been implemented successfully in the *Twin Burger vehicle* (Ishii *et al.*, 1995), however, the online implementation entails high computing power or parallel processing to obtain the controller parameters in real time. Aguiar and others (Oliveira *et al.*, 1998; Aguiar and Pascoal, 1997) have realised a

sliding mode controller in the *SIRENE* underwater vehicle based at Instituto Superior Técnico, Portugal. The controller provides good robustness to modelling errors and has essentially a PID type structure with additional nonlinear terms to account for vehicle parametric uncertainty. Another sliding mode control methodology was developed by Healey and Lienard (1993) at the Naval Post Graduate School (NPS), Monterey, CA. The NPS *ARIES* AUV is a successful example of the implementation of sliding mode technique to control the vehicle in the horizontal and vertical planes (Healey and Marco, 2001).

The next section elaborates on the hardware and software setup used for control system trials followed by the experimental results.

8.2 *Hammerhead* Setup for Control System Trials

Most of the initial *Hammerhead* testing has been performed by communicating with the vehicle through an umbilical attached to the rear end of the AUV. This is depicted in Figure 3.4 where it is shown that the umbilical is employed for sending command and control signals from a surface laptop which is communicating directly with the onboard sensors. In addition, a safety switch to turn the thruster on/off and a leak detector is also attached to the surface end of the cable which was being operated manually. A series of trials to identify the dynamic model of the vehicle for controller design evidently proved that the umbilical was causing significant amount of drag and extra weight on the AUV. In addition, the umbilical limits the vehicle operation since a surface platform was required to move along with the vehicle because of a finite cable length. This is similar to an ROV type setup except that the *Hammerhead* is powered from onboard batteries rather than the power being sent down via the umbilical. Hence the need to replace the cable is of paramount importance to test the autonomy of the whole system. A wireless ethernet link has thus been established between a surface laptop and an onboard host computer. The host computer runs a Labview visual interface which is used to gather all the sensory information and is then transmitted to an onboard controller machine when demanded. The controller is running in the Matlab/Simulink environment where an interface program decides which mission to execute based on the data string prefix. Both onboard CPUs are running Windows 2000 and the specifications are provided in Table 8.1.

	Host Computer	Controller Machine
Model	Pentium 4, 2.4 GHz	Pentium 4, 2.4 GHz
Memory	512 MB	512 MB
Storage	120 GB	40 GB
Operating System	Windows 2000	Windows 2000
Software	NI Labview	Matlab/Simulink
Other Hardware	NI DAQ and frame grabber card	USB to serial converters

Table 8.1: Onboard computer specifications

The use of image processing techniques in the host computer to obtain navigation information explains the high storage capacity required by the host CPU. A block diagram of the hardware setup within the *Hammerhead* AUV is depicted in Figure 8.1. This provides information on the flow of data between the onboard computers, actuators, sensors and the surface laptop.

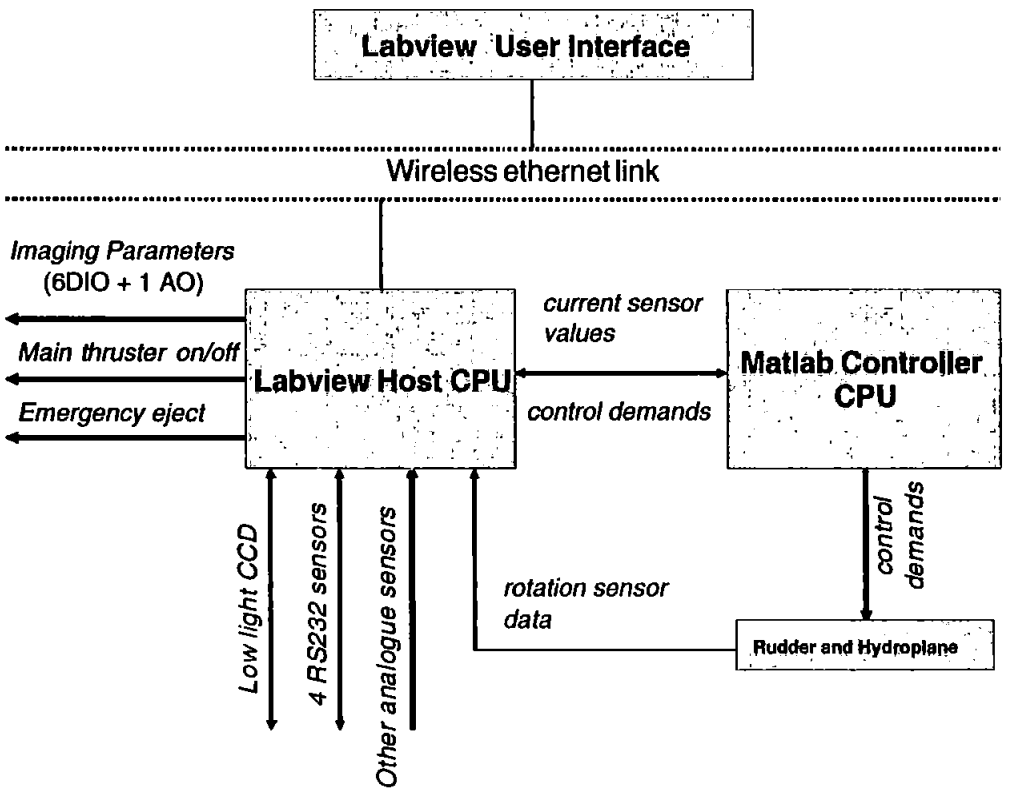


Figure 8.1: Information flow in *Hammerhead* AUV

As shown, the host computer transmits the sensor data in the form of a string to the Matlab machine. The controller CPU is directly manipulating the control surfaces through its own serial port. In addition to the sensors data, the string provides information on the mission type, the desired heading and depth to follow and a counter so that the Matlab data could be synchronised with the Labview data. A screenshot of the Labview visual interface running on the host machine which was developed at Cranfield University is depicted in Figure 8.2.

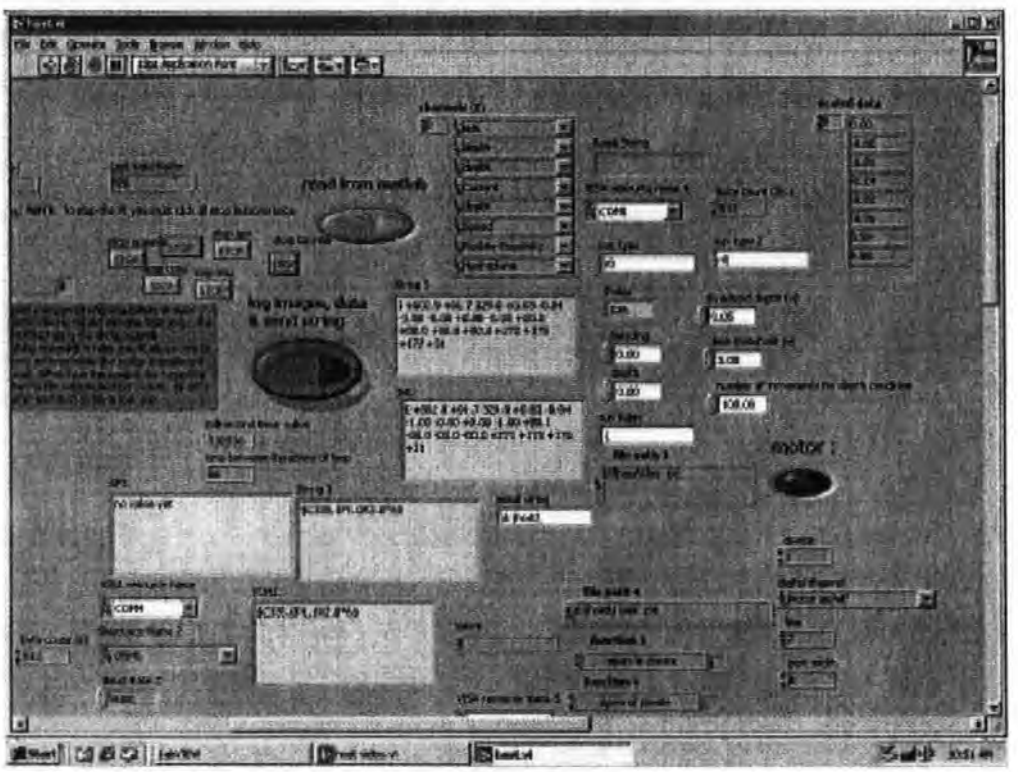


Figure 8.2: Screenshot of a Labview visual interface developed at Offshore Technology Centre, Cranfield University

All the sensors data can be seen in the figure including the desired heading and depth. In addition, the run type prefix can also be altered during a mission should there arise a need to terminate a mission prematurely. The controller CPU waits for an input command from the host computer. An *r0* string prefix establishes the communication between the two onboard machines and initialises the control planes. Another string prefix then dictates the mission to be executed. The data string received by the controller machine is of the following form

rX & IMU data & TCM2 data & GPS data & Hydoplane position & Rudder position

& Depth data & Counter & Desired heading & Desired depth
A sample string is also shown below

Sample String

r1 & I +001.7 -13.6 185.7 -0.23 -0.02 -0.97 -0.00 -0.00 ··· & $C183.1P-13.7R1.8 & $GPGGA, 162000, 5204.3658, N, 00037.6113, W, 0, ··· & 2.465820 & 1.469727 & 0.100000 & 250.000000 & 150.00 & 5.00

where rX dictates a mission to be executed followed by sensor strings and some other parameters to be used by the controller. Having decided a certain mission type, it is then executed while receiving feedback from onboard sensors. The host CPU now continuously transmits the data string with the same prefix. Figure 8.3 depicts the flowchart of the Matlab interface program. During each iteration of a specific mission run, certain checks are made to ensure safe operation. There are several conditions during the course of any mission, as shown in the flowchart, based on which a mission could be aborted. These are described below

If the run type is $r0$ This could be the case when the vehicle runs out of space or the data collected is assumed to be sufficient enough for further analysis. Operator intervention is required for this condition as currently there is no obstacle detection and avoidance module installed in the vehicle.

If the vehicle goes beyond a specified depth There are two different scenarios for this condition. For near surface missions where there is a need for a GPS fix, if the vehicle goes down below a prescribed depth due to some bow wave or some other malfunctioning, the mission controller terminates the program, shuts down the thruster and initialises the control surfaces so that the same mission could be re-executed, if needed. For underwater missions, if the vehicle goes beyond its maximum operating depth, then the same set of commands are issued as before. Owing to the positive buoyancy, the vehicle pops up on the surface in both cases.

Mission time is greater than the maximum allowable time This is directly related to the endurance of the *Hammerhead* vehicle. With all the sensors, computers and laser system onboard and running, it is envisaged that the maximum endurance of *Hammerhead* is approximately two hours. The maximum allowable mission time must therefore be less than the maximum endurance.

In the end, the program waits for the next mission type.

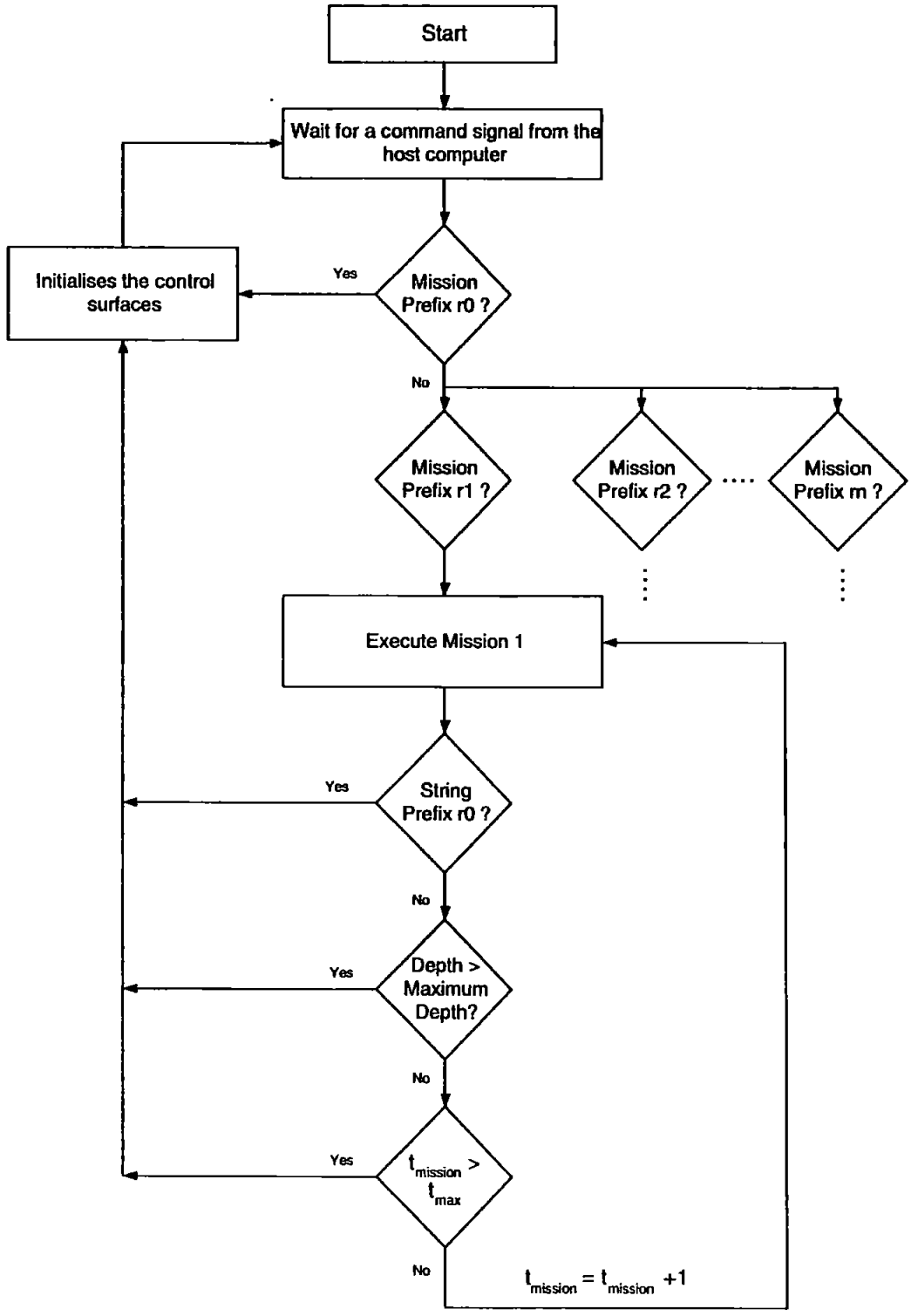


Figure 8.3: Information flow between onboard computers in *Hammerhead* AUV from a control system perspective

8.3 Experimental Results

Results are now presented of the application of a GA-MPC autopilot in the *Hammerhead*. As mentioned before, the SI trials have been conducted in Willen Lake, Milton Keynes for near surface work whilst the depth channel and controller trials were carried out at Roadford Reservoir, Devon. It is also important to point out the fact that the controllers were being tested on a 1960s made vehicle hull with the bulk of the existing electronics specifically the motors and their mechanical assembly being retained. Due to this, the rudder movement could not be controlled precisely and the minimum deflection observed was 2°. For this reason, a minimum rate of change of input constraint was imposed on the rudder which could lead to chattering effects in the rudder movement. The front hydroplanes which produces vertical motion are however new and there is no minimum rate constraint enforced on their movement.

8.3.1 GA-MPC autopilot results

For the application of this type of controller, there are two sets of trial results which need to be discussed. It was mentioned previously that the umbilical which was being used for communication purposes has been replaced by a wireless ethernet connection between an onboard host computer and a surface laptop. It is shown here that the controller is quite robust to cope with changes in vehicle dynamics resulting from the detachment of the umbilical. Simulation results for a step change in heading are first presented involving the 2° rate constraint. The output is then compared with the experimental data and some discussion is followed. Some of these results have already been published by the author (Naeem *et al.*, 2004*c*). The same model obtained from the umbilical based vehicle is then used to control the vehicle in the absence of the cable and results are shown to verify the robustness of the GA-MPC autopilot.

To simulate the controller, it is assumed that there is no plant/model mismatch. The constraints on the rudder were $\pm 20°$ while the minimum allowed deflection is 2° as discussed in the previous section. The weighting matrices were adjusted heuristically and the prediction and control horizons of MPC were set to minimise the control effort and to increase the speed of response. The tuning variables along with GA parameters used in the simulation are provided in Table 8.2.

Controller Parameters	Value
Q	1
R	0.5
S	0.0
H_p	7
H_c	1
Mutation prob.	0.005
Crossover prob.	0.7
No. of generations	10
Population size	100
Insertion rate	1

Table 8.2: GA-MPC tuning parameters for LOS tracking

The controller was simulated for a step change in heading and the result is depicted in Figure 8.4. Without taking any disturbance into account, the vehicle seems to be following the desired course closely after initiating from an arbitrary direction. The response in Figure 8.4(b) bears a small overshoot which can be minimised by adjusting the weighting matrices but at the cost of slower response time. The rudder deflections generated by the GA-based controller are also shown in Fig. 8.4(a) requiring minimum control effort and stays within the specified bounds. There is a large movement in the rudder position around $t = 50sec$ yet this does not affect the vehicle's motion because of its slow dynamics. The spike is due to the probabilistic nature of GA which produces such results unless accounted for in the code but has not been implemented herein owing to the extra computational burden that it imposes. The chattering phenomena can also be observed from this plot because of the 2° rate constraint. However, the effect of this is almost negligible on the vehicle's movement.

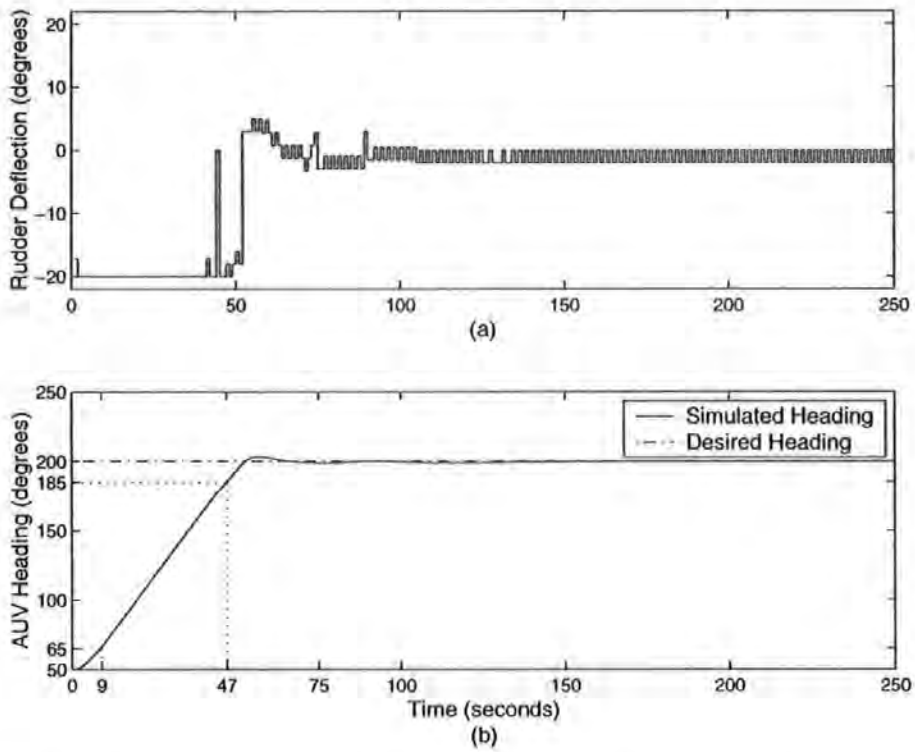


Figure 8.4: GA-MPC simulation results (a) Rudder deflections generated by the controller and (b) AUV heading

Next, the heading controller was tested in *Hammerhead*. The parameters used in these trials were kept the same as in the simulation studies for a fair comparison. The test was carried out for a step change in heading. The initial and desired headings were kept the same and the vehicle was allowed to swim freely. The result from the first in water test is depicted in Figure 8.5. As shown, the test could not be completed as the vehicle ran out of space and the mission had to be aborted. Although incomplete, the vehicle seems to be locked up onto the desired target, Figure 8.5(b). However, the run time was not enough to examine the effects of model discrepancies which could show up as an offset to the desired heading or no tracking at all. The control surface as seen in Figure 8.5(a) is also settling down to its steady state value as the vehicle continue its tracking mission.

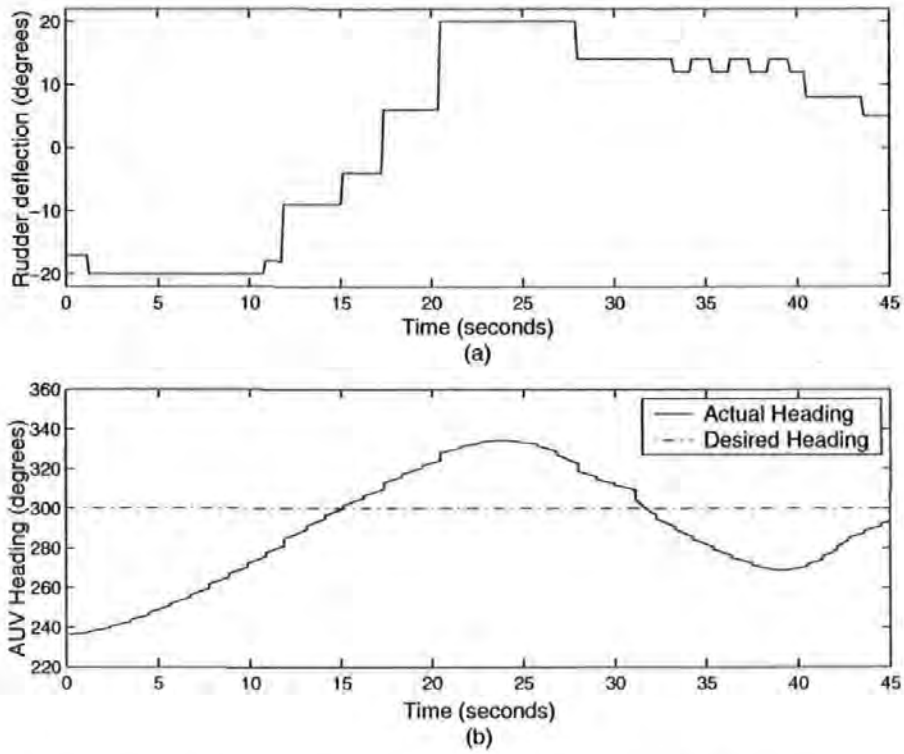


Figure 8.5: Controller trial results data set 1 with umbilical (a) Rudder deflections generated by the GA-MPC (b) *Hammerhead* heading obtained from an onboard IMU

The same experiment was then recapitulated for a longer duration and the results are shown in Figure 8.6. It is evident from Figure 8.6(b) that the GA-MPC was able to track the desired heading without any offset despite the presence of always existing model uncertainty and external disturbances. There is an overshoot though which could be blamed on the surface currents. Looking at the rudder deflections in Figure 8.6(a), again there is some expected chattering present. However, the rudder movement always remains within the specified constraints. There is a large spike followed by fluctuations in rudder movement at approximately 125 seconds in response to the change in vehicle's heading due to surface currents. However, the controller is robust enough to cope with it and attains the steady state input and output values in approximately 50 seconds. A statistical analysis reveals that the standard deviation of rudder deflections in experimental data is approximately 11°, while it is quite high (8°) in the simulation (even without any disturbance), possibly because of the chattering phenomena.

It is interesting to note that the rise time in the experimental data is much smaller

(24 sec) than in the simulation (38 sec). One reason for this is the effect of surface currents pushing the vehicle unwantingly and causing even a higher overshoot as compared to the simulated response. Model/plant mismatch could also be another source of this problem.

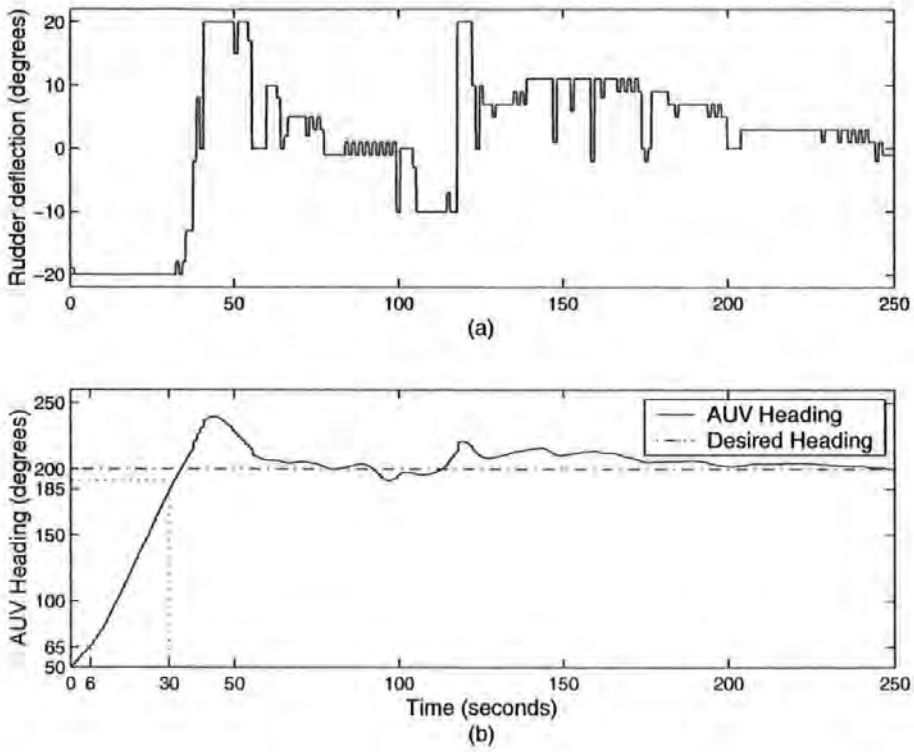


Figure 8.6: Controller trial results data set 2 with umbilical (a) Rudder deflections generated by the GA-MPC and (b) *Hammerhead* heading obtained from an onboard IMU

Experiments were next performed with an umbilical free vehicle hence could be termed as one of the first autonomous runs of *Hammerhead* AUV. The surface laptop was only used to initiate a mission onboard the host computer through a wireless connection and then let the vehicle swim on its own to complete the task. The controller parameters including the GA variables are provided in Table 8.3.

The result of a LOS tracking mission is depicted in Figure 8.7(a). The vehicle initial heading is arbitrary whereas the heading to be attained was selected as 150^0. This time the vehicle takes the shortest path in the counter clockwise direction. Clearly, the length of the experiment is again not sufficient enough to gauge the tracking capabilities and the effect of disturbances on the AUV motion. However, the results provide evidence that the GA-MPC is quite robust in tackling the modelling uncer-

Controller Parameters	Value
Q	1
R	0.5
S	0.0
H_p	7
H_c	1
Crossover prob.	0.7
No. of generations	10
Population size	100
Insertion rate	1

Table 8.3: GA-MPC tuning parameters for LOS tracking during the first autonomous run

tainty introduced by the absence of the umbilical. The vehicle seems to be following the desired heading closely without an overshoot and offset. Moreover, the settling time is approximately 65 seconds which is also quite close to the previous case. A plot of the rudder deflections generated by the GA-MPC autopilot is also depicted in Figure 8.7(b).

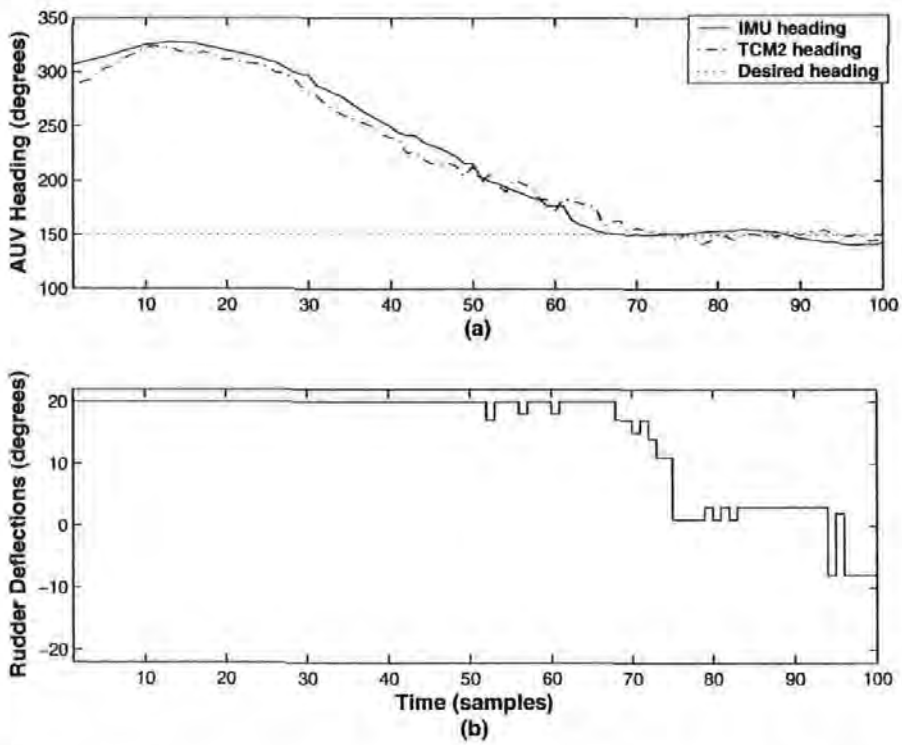


Figure 8.7: First autonomous *Hammerhead* trial results using GA-MPC autopilot without umbilical (a) Rudder deflections generated by the controller (b) AUV heading

The 2° chattering effect is evident, however, the rudder movement is at all times stays within the specified actuator limits and attains its steady state value as the vehicle approaches the desired LOS. Figure 8.8 unveils other sensory information obtained from an onboard IMU and pressure sensor. All three plots show transient behaviour in the beginning when the thruster started and then settled to their steady state values after some time. The trials were performed near to the surface hence the depth output stays close to zero throughout. Since no vertical motion was produced by the AUV, therefore the pitch remains at 0°. There is some movement observed in the roll data which is due to the presence of strong surface currents generated by strong winds on that particular day. However, this does not influenced the heading data significantly.

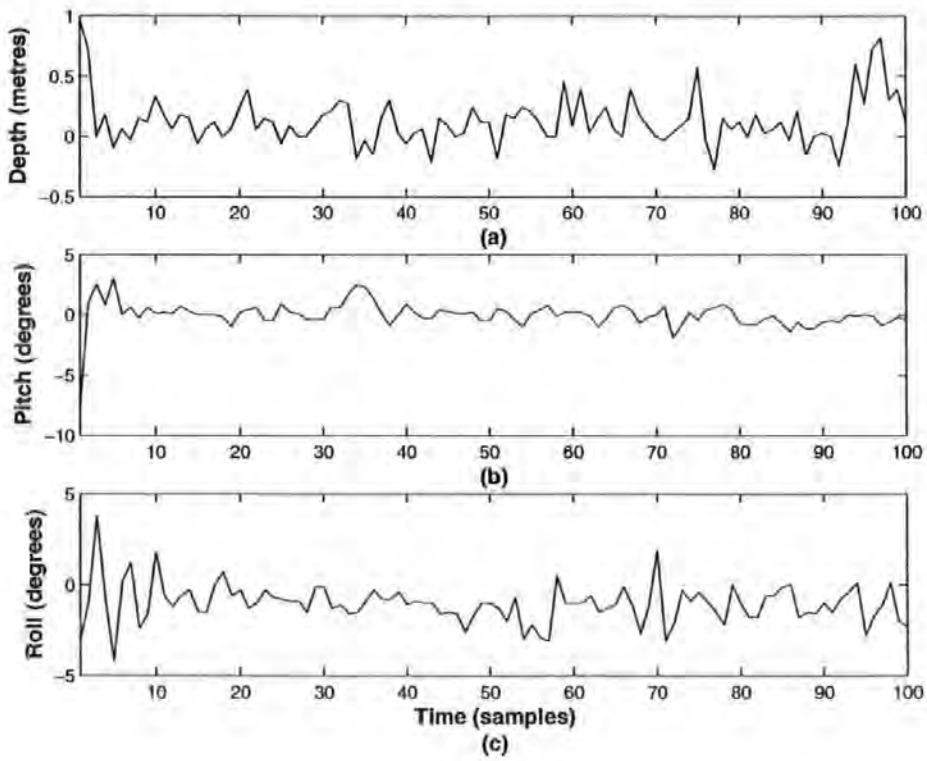


Figure 8.8: First autonomous *Hammerhead* trial results using GA-MPC autopilot without umbilical showing (a) Depth (b) Pitch and (c) Roll

Finally, the *Hammerhead* trajectory is shown in Figure 8.9 during the first autonomous run. The longitude and latitude data has been obtained from an onboard GPS and converted into position coordinates in body fixed frame of reference. The vehicle was launched from the origin and it covers a maximum displacement of approximately 30 and 6 metres in the x and y directions respectively. The AUV initial

trajectory is a straight line in the negative x-direction for the first few metres. This is because the autopilot was gathering the data and initialising the variables for future calculations and hence no input was applied to the rudder during this time. The vehicle first made a turn in the negative y-direction and travelled about 10 metres before making a right turn in the positive y-direction. The kinks evident in the plot are possibly due to the missing data or it could be because of the inaccuracy of the GPS sensor.

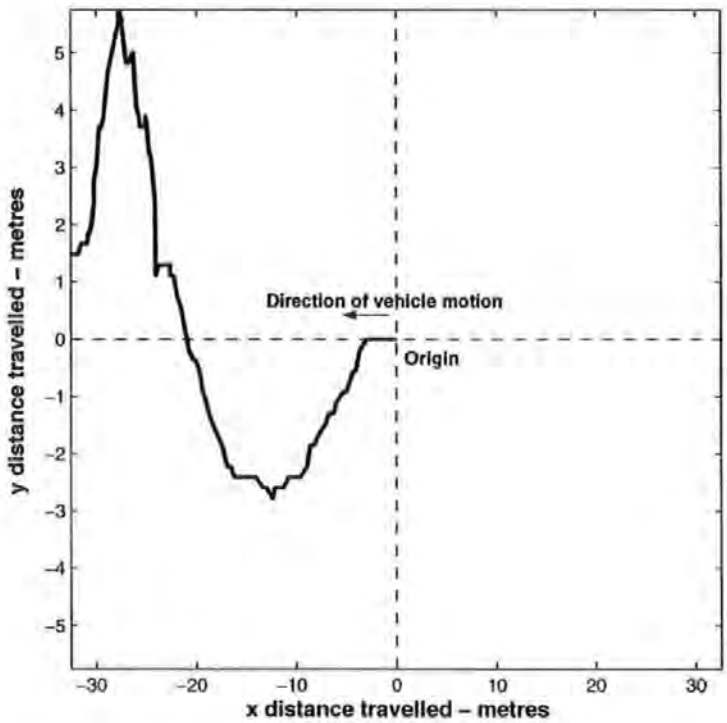


Figure 8.9: GPS data acquired during the first autonomous autopilot run showing *Hammerhead* trajectory in body coordinate frame in a LOS tracking mission

8.4 Concluding Remarks

The final logical step in a control system development is to gauge the capabilities of the controller in real time. Simulations can provide a good insight into the vehicle's behaviour, however, the true potential can only be judged through experimentation. In this chapter, results of the application of a GA-MPC autopilot in the *Hammerhead* AUV have been revealed. It is shown that the controller is capable of producing outstanding results given different vehicle configurations (with and without umbilical), albeit the model was extracted using the data obtained from an umbilical

based AUV. The results demonstrated the robustness of the GA-MPC autopilot under varying conditions such as environmental effects and it cope well with the modelling uncertainty.

Unfortunately, the experiments involving the LQG/LTR and MPC with a fuzzy objective function could not be commissioned within due time and hence the results were not available by the time of the completion of this thesis. It is envisaged that given the robustness of these controllers in simulation environment, they will perform well in real time in the *Hammerhead* AUV.

Chapter 9

Summary, Conclusions and Recommendations for Further Work

This closing chapter contains the summary of the work presented within the thesis, conclusions and recommendations for further work. This includes a brief summary of each of the chapters followed by conclusions drawn which are with respect to what was anticipated and what has been achieved in terms of the aim and objectives of the research programme. Lastly, recommendations for the future work are provided that could stimulate further research projects in this field of study.

9.1 Summary

AUV research and development is on the verge of reaching maturity yet applications are very few. The cost associated with an AUV development particularly of the on-board sensors, power requirements and underwater testing have imposed a significant constraint on its development. Numerous ideas regarding underwater vehicle control have been proposed in the literature however the lack of test equipment confines one to simulations only. The aim of this project as defined in Chapter 1 was to produce a low cost prototype AUV which could be exploited as a testbed by other underwater research groups within the UK. To satisfy this requirement, the vehicle command and control system should be flexible enough to accommodate various requirements set by the user.

The vehicle named *Hammerhead* has been commissioned as a cooperative project by the Universities of Plymouth and Cranfield. In addition to the control system, an MSDF based navigation algorithm has also been formulated at Plymouth which is based on AI techniques such as fuzzy logic and genetic algorithms. To enhance the capability of the navigation system and hence the MSDF, a laser stripe illumination based vision system was developed at Cranfield which provides accurate altitude and velocity information using a combination of an onboard camera and laser stripe. A signal processing algorithm is then employed to extract the necessary information.

This thesis presents several aspects of an underwater vehicle design from the laboratory stage to operation under water. Overall, the thesis has contributed significantly to the underwater research literature by disseminating the research work carried out within. In summary, the primary aim of this research alone was to develop guidance and control laws based on advance control concepts which are suitable for underwater operation. The implementation of these controllers in *Hammerhead* was imperative to gauge their robustness and hence render the vehicle fully autonomous. Experiments have been carried out to obtain data sets whereby SI techniques were suggested and applied to model the *Hammerhead* yaw and depth channel dynamics. Two novel form of guidance laws were also proposed which are designed for cruising type vehicles like *Hammerhead*. The control systems developed in this thesis are a combination of optimal control and AI. The LQG/LTR controller is selected because of its remarkable stability and robustness properties whereas the MPC autopilot has been developed because of its strong robustness and constraint handling characteristics. Simulation results were shown in Chapters 6 and 7 for several scenarios. The guidance laws proposed in Chapter 5 were integrated with the autopilots and the results have been compared. In addition, simple guidance strategies such as LOS and waypoint following schemes were also integrated with the proposed controllers and results have been demonstrated for various settings. Moreover, the control systems were simulated in conjunction with a vision based altitude information sensor and results demonstrate excellent robustness and disturbance rejection characteristics. The thesis also reports original results on the implementation of an online GA to control the yaw angle of the *Hammerhead* in Chapter 8. These results are significant since to the author's knowledge, this is the first known application of an online GA in the marine sector. The next section provides a brief summary of the individual chapters, work carried out therein and the conclusions drawn.

9.2 Conclusions

Chapter 2 provided comprehensive background material on guidance systems design for autonomous vehicles with an emphasis on UUVs. Firstly, missile guidance laws were explored which was then followed by guidance strategies for underwater vehicles. It was noticed that a majority of AUVs focus on vision based guidance, however, the performance of these types of system degrade when subjected to high water turbidity and low light conditions. Electromagnetic guidance is more efficient but the applications are quite limited. Another alternative is the use of a sonar which is unaffected by the above mentioned problems. In summary, it was concluded from the discussion presented that the LOS guidance in one form or the other is the key constituent of most guidance strategies. It should be noted that the material forming this chapter has been disseminated as a review paper (Naeem *et al.*, 2003*d*).

The *Hammerhead* hardware configuration was next detailed in Chapter 3. The chapter demonstrated the overall physical structure of the *Hammerhead* vehicle as it evolved from an ROV type setup i.e., communicating via an umbilical, to its fully autonomous conformation. The navigation sensors were briefly reviewed and actuator control design was explicated along with its hardware and software which is currently being used in the *Hammerhead*. This was designed in a flexible way so that a user familiar with any software language with a serial interface could take advantage of it.

Chapter 4 explored the dynamic characteristics of the *Hammerhead* vehicle itself. A general description of the mathematical modelling techniques was presented followed by a terse review on SI. Trials were carried out to collect data sets for the purpose of extracting yaw and depth channel models of the AUV using SI. The dynamic models were verified by independent data sets and results were shown extensively. It has been demonstrated that the models obtained using SI are adequate for controller design purposes and avoids painstaking mathematical modelling which involves tank tests to estimate the hydrodynamic coefficients.

Some new guidance strategies for *Hammerhead* vehicle have been proposed in Chapter 5. The guidance laws were specifically designed for underwater cable inspection tasks for cruising type vehicles. The first guidance strategy presented has evolved from airborne systems and was termed as pure pursuit guidance. In the other guidance law, the vehicle speed has been used as a means to formulate the guidance scheme.

This was called a hybrid guidance system since it utilises various existing guidance laws during different phases of the mission. Simulation results were shown for both guidance strategies and the performances were evaluated. From an implementation perspective, it is currently not possible to test the hybrid guidance law in *Hammerhead* since its dynamic model is only available at one fix speed. However, the concept has been verified by utilising an AUV dynamic model from the literature which is represented in terms of vehicle velocity. It was shown that both guidance systems are capable of producing suitable reference trajectories to be followed by the vehicle in order to accomplish a cable tracking objective. Literature have been published by incorporating the pure pursuit guidance and hybrid law with a GA-MPC autopilot developed in Chapter 7 (Naeem *et al.*, 2004*b*) and LQG/LTR controller in Chapter 6 (Naeem *et al.*, 2003*a*) respectively.

A discrete-time LQG/LTR controller was formulated next in Chapter 6 for *Hammerhead* vehicle models in the lateral and dive planes. The LQG/LTR autopilot was integrated with basic guidance laws such as LOS, waypoint guidance and more advanced strategies proposed in Chapter 5. Simulation results for all cases were shown including the effects of sea current disturbances. A vision based altitude information system has also been integrated with the controller and results showed excellent tracking behaviour and robustness in the presence of disturbances and uncertainties in the input demand. The LQG/LTR was proved to be quite robust under different operating conditions. However, a major disadvantage is that several parameters need to be tuned and a tedious frequency domain analysis has to be carried out before an acceptable design could be obtained. Part of the material in this chapter has been utilised to publish a paper by applying the LQG/LTR autopilot to the *Hammerhead* models extracted in Chapter 4 (Naeem *et al.*, 2003*b*).

Chapter 7 demonstrated successful controller design based on an MPC control strategy integrated with a GA and fuzzy logic. Two modifications have been suggested to the standard predictive control problem. The first technique replaced the conventional optimizer with a GA which has the advantage of generalising the MPC to use any type of process model and cost function. The second modification was to replace the conventional quadratic cost function with a fuzzy performance index. A GA being employed again as an optimization tool. The fuzzy cost function was shown to be more intuitive than a quadratic cost function. Other advantages of using fuzzy logic is the automatic implementation of hard and soft constraints, normalisation and ease

of tuning. Integration of both type of controllers with the standard and proposed guidance systems were illustrated. Simulation results demonstrated the efficacy of the proposed techniques in the presence of sea currents and when the input demand from the guidance system is uncertain. Overall, the proposed autopilots were proved to be quite efficient as compared to the LQG/LTR controller. In addition, for the hybrid guidance system, a single GA-MPC controller was tuned which was employed for all models and results were shown to be much more improved than obtained using the LQG/LTR control strategy. The work involving GA-MPC to control an AUV has been disseminated by the author (Naeem, 2002; Naeem *et al.*, 2004*b*; Naeem *et al.*, 2003*c*; Dalgleish *et al.*, 2004) whereas the results involving the use of fuzzy logic in an MPC framework was published in Naeem *et al.* (2004*a*).

Finally, the implementation of one of the proposed autopilots, namely the GA-MPC in *Hammerhead* has been undertaken in Chapter 8. This is considered to be one of the principal novelties of this research where an online GA is employed for optimization in an MPC framework. Results from actual in water experiments were demonstrated and the performance was compared with simulation examples. It was shown that the controller is quite robust under a varying set of conditions and it is capable of dealing with the ever present modelling uncertainty resulting from the removal of the umbilical. Unfortunately, the implementation of the remaining autopilots could not be carried out and hence no results were available by the time of the completion of this thesis.

9.3 Recommendations for Future Work

The research work presented within this thesis has the honour to be the first reported study of the *Hammerhead* AUV dynamics and controller design. A number of achievements have been made throughout the course of this thesis. However, scientific research is an ongoing process and clearly there are numerous avenues for future research and development involving *Hammerhead*. Some of the work is quite perceptible and could not be accomplished because of the lack of time. A list of potential follow ups based on this work are provided below.

- The first and most obvious future work to be followed is to carry out further experimentation involving the remaining autopilots which have not yet been tested in the *Hammerhead*. Furthermore, for the GA-MPC controller, the depth control experiments could not be accomplished in addition to the integration

with other guidance strategies such as waypoint following. Similar trials for the LQG/LTR and MPC using fuzzy cost function could also be conducted and a qualitative and quantitative comparison should be made.

- The generic AUV control design approach is to use mathematical modelling to generate a nonlinear six DOF model of the vehicle which is then linearised around some operating point. However, in this thesis, open loop SI techniques have been applied to extract the linear dynamic models of the *Hammerhead* AUV. Further studies of *Hammerhead* could involve developing a mathematical model of the vehicle based on laws of physics. The linearised models could then be compared with the models obtained using SI. Another approach is to employ closed loop SI, in particular, to the depth dynamics where open loop experiments are not adequate to obtain ample dynamical information and multiple experiments need to be carried out. The same data pairs obtained for SI could also be utilised to estimate a nonlinear model of the vehicle. Neural networks are a potential tool for a nonlinear model development using the existing *Hammerhead* data. It was mentioned in Chapter 7 that the GA-MPC autopilot can handle any type of process model and thus the true potential of the proposed algorithm could then be exploited.

- The models and controllers developed in this thesis are SISO assuming that there is no strong cross coupling present between various channels. However, it is worth formulating a linear or nonlinear MIMO model and hence a MIMO autopilot and compare the performance with the SISO controllers. The LQG/LTR and GA-MPC autopilots developed in this thesis have a distinct advantage of producing MIMO controllers as they are inherently multivariable. Hence the extension from SISO to MIMO is rather straightforward.

- It was mentioned in Chapter 4 that some cross-coupling has been observed between different channels. In particular, the depth and pitch channel shows the strongest coupling. Attention could be focussed on developing a model of the AUV that takes into account this cross coupling effect. An autopilot could then be developed to maintain the depth by controlling the pitch of the vehicle. This could be achieved by introducing a positive pitch angle until the vehicle approaches the specified depth. The pitch is then gradually decreased so that the AUV lines up with the desired depth and maintains the level at zero pitch angle.

References

Aguiar, A. and A. Pascoal (1997). Modelling and control of an autonomous underwater shuttle for the transport of benthic laboratories. In: *Proceedings Of Oceans'97.* MTS/IEEE. Halifax, Canada. pp. 888–895.

Aguiar, A., P. Oliveira, C. Silvertre and A. Pascoal (1998). Guidance and control of the SIRENE underwater vehicle: from system design to tests at sea. In: *Proceedings Of Oceans'98.* Vol. 1-3. Nice, France. pp. 1043–1048.

Ahmad, S. M. and R. Sutton (2003). Dynamic modelling of a remotely operated vehicle. In: *Proceedings 1st IFAC workshop on guidance and control of underwater vehicles GCUV2003.* IFAC. Newport, South Wales, UK. pp. 47–52.

Ahmad, S. M., R. Sutton and R. S. Burns (2003). Retrieval of an autonomous underwater vehicle: An interception approach. *Journal of the Society for Underwater Technology* **25**(4), 185–197.

Akkizidis, I. S., G. N. Roberts, P. Ridao and J. Batlle (2003). Designing a fuzzy-like PD controller for an underwater robot. *Control Engineering Practice* **11**(4), 471–480.

Asakawa, K., J. Kojima, Y. Ito and S. Takagi (1996). Autonomous underwater vehicle AQUA EXPLORER 1000 for inspection of underwater cables. In: *Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology.* IEEE. Monterey, CA, USA. pp. 10–17.

Bakaric, V., Z. Vukic and R. Antonic (2004). Improved line of sight guidance for cruising underwater vehicles. In: *Proceedings of Control Applications in Marine Systems (CAMS'04).* IFAC. Ancona, Italy. pp. 447–452.

Balasuriya, A. and T. Ura (1998). Autonomous target tracking by underwater robots based on vision. In: *Proceedings of IEEE UT'98.* Tokyo, Japan. pp. 191–197.

Balasuriya, A. and T. Ura (1999*a*). Multisensor fusion for autonomous underwater cable tracking. In: *Oceans' 99.* MTS/IEEE. Seattle, WA, USA. pp. 209–215.

Balasuriya, A. and T. Ura (1999*b*). Sensor fusion technique for cable following by autonomous underwater vehicles. In: *Proceedings of the 1999 IEEE International Conference on Control Application.* IEEE. Kohala Coast-Island of Hawai'I, USA. pp. 1779–1784.

Balasuriya, A. and T. Ura (2000). Autonomous target tracking by twin-burger 2. In: *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE/RSJ. Kagawa University, Takamatsu, Japan. pp. 849–854.

Balasuriya, A. and T. Ura (2001). Underwater cable following by twin-burger 2. In: *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*. IEEE. Seoul, Korea. pp. 920–925.

Bannon, R. T. (1998). Rovs and undersea cable maintenance. In: *Proceedings of the Underwater Technology' 98*. Tokyo, Japan.

Barber, M. (2001). *Hydrographic crew surveys underwater quake damage*. World Wide Web, Date Accessed, August 16th 2004. `http://seattlepi.nwsource.com`.

Bjerrum, A. and T. Slater (2001). Autonomous tracking of submarine pipelines and cables. In: *Proceedings of Hydro 2001, Hydrographic Society*. Norwich, UK.

Bossley, K. M., M. Brown and C. J. Harris (1999). Neurofuzzy identification of an autonomous underwater vehicle. *International Journal of System Science* **30**(9), 901–913.

Briest, S., S. Cowen and J. Dombrowski (1997). Underwater docking of autonomous undersea vehicles using optical terminal guidance. In: *Oceans'97 MTS/IEEE Conference Proceedings*. Vol. 1-2. Halifax, Canada. pp. 1143–1147.

Brown, J. M., D. B. Ridgely and R. N. Paschall (1994). Autopilot design for a tail-controlled missile using LQG/LTR with eigenstructure reassignment. In: *Proceedings of the 1994 American Control Conference*. AACC. Baltimore, MD, USA. pp. 3278–3282.

Brown, R. G. and P. Hwang (1997). *Introduction to Random Signals and Applied Kalman Filtering*. Wiley and Sons, Inc.. New York, USA.

Bryson, A. E. and Y. C. Ho (1975). *Applied Optimal Control*. Halsted Press, Washington, D.C.

Burl, J. B. (1999). *Linear Optimal Control, H_2 and H_∞ Methods*. Addison-Wesley Longman Inc.

Caccia, M., G. Bruzzone and G. Veruggio (2000). Guidance of unmanned underwater vehicles: experimental results. In: *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*. IEEE. San Francisco, USA. pp. 1799–1804.

Carlos, E. G., D. M. Prett and M. Morari (1989). Model predictive control: Theory and practice - a survey. *Automatica* **25**(3), 335–348.

Carlowicz, M. (2003). *Jason and the Golden Fleece, The Epic Quest of Woods Hole's ROV Engineers.* World Wide Web. Date Accessed, July 27th 2004. `http://www.whoi.edu/home`.

Chen, C. T. (1984). *Linear System Theory and Design.* CBS College Publishing.

Chen, G. and S. H. Chung (1994). Depth control of a submersible vehicle under a seaway using a noise estimator. In: *Proceedings of the 6th International Conference on Electronic Engineering in Oceanography.* IEE. Cambridge, UK. pp. 30–34.

Clarke, D., Ed.) (1994). *Advances in Model-Based Predictive Control.* Oxford science publications.

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987*a*). Generalised predictive control. Part 1: The basic algorithm. *Automatica* **23**(2), 137–148.

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987*b*). Generalised predictive control. Part 2: Extensions and interpretations. *Automatica* **23**(2), 149–160.

Cloutier, J. R., J. H. Evers and J. J. Feeley (1989). Assessment of air-to air missile guidance and control technology. *IEEE Control Systems Magazine* **9**(6), 27–34.

Consi, T. R., C. A. Goudey, J. Cho, J. Atema and C. Chryssostomidis (1994). AUV guidance with chemical signals. In: *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology.* IEEE. Cambridge, MA. pp. 450–455.

Craven, P. J., R. Sutton and R. S. Burns (1998). Control strategies for unmanned underwater vehicles. *The Journal of Navigation* **51**(1), 79–105.

Craven, P. J., R. Sutton and R. S. Burns (1999). Multivariable neurofuzzy control of an autonomous underwater vehicle. *Integrated Computer-Aided Engineering* **6**(4), 275–288.

Creaser, P. A., B. A. Stacey and B. A. White (1998). Evolutionary generation of fuzzy guidance laws. In: *UKACC International Conference on Control' 98.* Vol. II. UKACC. UK. pp. 883–888.

Cutler, C. R. and B. L. Ramaker (1980). Dynamic matrix control - a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference.* San Francisco, CA, USA.

Dalgleish, F. R. (2001). An experimental study into the dynamic behaviour of the *Hammerhead* AUV. Master's thesis. School of Industrial and Manufacturing Science, Cranfield University. Cranfield, Beds, UK.

Dalgleish, F. R., S. Tetlow and R. L. Allwood (2002). The design of a laser-based imaging sensor for AUV navigation (poster presentation). In: *IOP PHOTON 02 Exhibition.* Cardiff, Wales, UK.

Dalgleish, F. R., S. Tetlow and R. L. Allwood (2003). Preliminary experiments in the development of a laser-based imaging sensor for AUV navigation. In: *Proceedings GCUV 2003*. Newport, Wales, UK.

Dalgleish, F. R., W. Naeem, S. Tetlow, R. Allwood and R. Sutton (2004). Adaptive AUV control for optimized swathe laser stripe imaging. In: *Proceedings of the 6th International Conference on Mechatronics*. IFAC. Sydney, Australia. pp. 175–180.

D'Azzo, J. J. and C. H. Houpis (1995). *Linear Control System Analysis & Design, Conventional and Modern*. McGraw-Hill Education (ISE Editions). USA.

Deshpande, P. B., Ed.) (1989). *Multivariable Process Control*. Instrument Society of America.

Duwaish, H. and W. Naeem (2001). Nonlinear model predictive control of hammerstein and wiener models using genetic algorithms. In: *Proceedings of the 2001 IEEE International Conference on Control Applications (CCA01)*. IEEE. Mexico City, Mexico. pp. 465–469.

Feezor, M. D., F. Y. Sorrell, P. R. Blankinship and J. G. Bellingham (2001). Autonomous underwater vehicle homing/docking via electromagnetic guidance. *IEEE Journal of Oceanic Engineering* **26**(4), 515–521.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley & Sons.

Franklin, G. F., J. D. Powell and M. Workman (1998). *Digital Control of Dynamic Systems*. 3rd ed.. Addison-Wesley Longman Inc.

Garnell, P. (1980). *Guided Weapon Control Systems*. 2 ed.. Brassey's Defence Publishers.

Goheen, K. R. and E. R. Jefferys (1990). The application of alternative modelling techniques to ROV dynamics. In: *Proceedings IEEE International Conference on Robotics and Automation*. IEEE. Cincinnati, OH, USA. pp. 1302–1309.

Grasso, F. W. (2001). Invertebrate-inspired sensory-motor systems and autonomous, olfactory-guided exploration. *The Biological Bulletin 200*. pp. 160–168.

Guo, J., F. C. Chiu and C. C. Huang (2003). Design of a sliding mode fuzzy controller for the guidance and control of an autonomous underwater vehicle. *Ocean Engineering* **30**(16), 2137–2155.

Hamilton, Kelvin (2004). *RAUVER MkII, Autonomous Hover-Capable Intervention and Close Inspection Vehicle*. World Wide Web. Date Accessed, August 11th 2004. `http://www.ece.eps.hw.ac.uk/~kelvin/oceansweb/rauver/rauver_index_mk2.htm`.

Healey, A. J. and D. B. Marco (2001). Command, control and navigation: Experimental results with the NPS ARIES AUV. *IEEE Journal of Oceanic Engineering, Special Issue on Autonomous Ocean Sampling Networks* **26**(4), 466–477.

Healey, A. J. and D. Lienard (1993). Multivariable sliding model control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering* **18**(3), 327–339.

Ippoliti, C. G., S. Radicioni and A. Rossolini (2001). Multiple models control of a remotely operated vehicle: Analysis of models structure and complexity. In: *Proceedings IFAC conference on control applications in marine systems, CAMS'2001.* IFAC. Glasgow, Scotland, UK.

Ishii, K., T. Fujii and T. Ura (1995). An on-line adaptation method in a neural network based control system for AUVs. *IEEE Journal of Oceanic Engineering* **20**(3), 221–228.

Ito, Y., N. Kato, J. Kojima and S. Takagi (1994*a*). Cable tracking for autonomous underwater vehicle. In: *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology.* IEEE. Cambridge, MA. pp. 218–224.

Ito, Y., N. Kato, J. Kojima and S. Takagi (1994*b*). Cable tracking for autonomous underwater vehicle. In: *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology.* IEEE. pp. 218–224.

Jones, Emma G., A. Tselepides, P.M. Bagley and I.G. Priede (2004). *Behaviour of Deep-Sea Sharks Inferred from in Situ Acoustic Tracking in the Cretan Sea, Eastern Mediterranean.* World Wide Web. Date Accessed, August 11th 2004. `http://www-heb.pac.dfo-mpo.gc.ca/congress/2002/DeepSea/Jones.pdf`.

Jordan, K. (2003). *REMUS AUV Plays Key Role in Iraq War.* World Wide Web. Date Accessed, July 27th 2004. `http://www.diveweb.com/rovs/features/034.01.htm`.

Juul, D. L., M. McDermott, E. L. Nelson, D. M. Barnett and G. N. Williams (1994). Submersible control using the linear quadratic gaussian with loop transfer recovery method. In: *Proceedings of the IEEE Symposium on Autonomous Underwater Technology.* Cambridge, MA, USA. pp. 417–425.

Kato, N., Y. Ito, J. Kojima, S. Takagi, K. Asakawa and Y. Shirasaki (1994*a*). Control performance of autonomous underwater vehicle AQUA EXPLORER 1000 for inspection of underwater cables. In: *Proceedings of Oceans Engineering for Today's Technology and Tomorrow's Preservation.* Vol. 1. IEEE. Brest, France. pp. I/135–I/140.

Kato, N., Y. Ito, J. Kojima, S. Takagi, K. Asakawa and Y. Shirasaki (1994*b*). Control performance of autonomous underwater vehicle "AQUA EXPLORER 1000" for inspection of underwater cables. In: *Proceedings of Oceans Engineering for Today's Technology and Tomorrow's Presentation.* IEEE. pp. I/135–I/140.

Kim, K. and T. Ura (2003). Fuel-optimally guided navigation and tracking control of AUV under current interaction. In: *Proceedings of Oceans Conference*. Vol. 2. IEEE. San Diego, CA, US. pp. 663–670.

Kuipers, J. B. (1999). *Quaternions and Rotation Sequences*. Princeton University Press, Princeton, New Jersey.

Kwiesielewicz, M., W. Piotrowski and R. Sutton (2001). Predictive versus fuzzy control of autonomous underwater vehicle. In: *Proceedings IEEE International Conference on Methods and Models in Automation and Robotics*. IEEE. Miedzyzdroje, Poland. pp. 609–612.

Lea, Roy Kim (1998). Control of a Tethered Underwater Flight Vehicle. PhD thesis. University of Southampton. UK.

Lennox, B., H. Zhang, D. Lovett and D. Sandoz (2004). An integrated approach to advanced process control and condition monitoring. *IEE Computing and Control Engineering Magazine* **15**(1), 32–37.

Li, J. H., P. M. Lee and S. J. Lee (2002). Neural net based nonlinear adaptive control for autonomous underwater vehicle. In: *Proceedings of the IEEE International Conference on Robotics and Automation, v 2*. Washington DC, USA. pp. 1075–1080.

Lin, C. F. (1991). *Modern Navigation, Guidance and Control Processing*. Vol. II. Prentice Hall.

Lin, C. L. and H. W. Su (2000). Intelligent control theory in guidance and control system design: an overview. *Proceedings of the National Science Council, Part A, Physical Science and Engineering, Republic of China, (Invited Review Paper)* **24**(1), 15–30.

Lin, J. M., C. H. Chang and H. L. Tsai (1997). Optimal feedforward and feedback controllers designed by LQG/LTR method. *Journal of the Chinese Institute of Engineers, Transactions of the Chinese Institute of Engineers, Series A/Chung-kuo Kung Ch'eng Hsuch K'an* **20**(1), 57–66.

Ljung, L. (1999). *System Identification, Theory for the User*. 2nd ed.. Prentice Hall PTR.

Ljung, L. (2001). *System Identification Toolbox User's Guide, Version 5*. The MathWorks Inc.

Locke, A. S. (1955). *Guidance. D. Van Nostrand Co. Inc.*. Princeton, NJ.

Loebis, D., F. R. Dalgleish, R. Sutton, S. Tetlow, J. Chudley and R. L. Allwood (2003). An integrated approach in the design of a navigation system for an AUV. In: *Proceedings of MCMC2003*. Girona, Spain. pp. 319–324.

Loebis, D., R. Sutton and J. Chudley (2002). Review of multisensor data fusion techniques and their application to autonomous underwater vehicle navigation. *IMarEST Journal of Marine Engineering and Technology* **A**(1), 3–14.

Loebis, D., R. Sutton, J. Chudley and W. Naeem (2004). Adaptive tuning of a Kalman filter via fuzzy logic for an intelligent AUV navigation system. *Control Engineering Practice* **12**(12), 1531–1539.

Maciejowski, J. M. (1985). Asymptotic recovery for discrete-time systems. *IEEE Transactions on Automatic Control* **AC-30**(6), 602–605.

Maciejowski, J. M. (2002). *Predictive Control with Constraints*. Prentice hall.

Mamdani, E. H. and S. Assilian (1975). An experiment in linguistic synthesis of fuzzy controllers. *International Journal of Man-Machine Studies* **7**, 1–13.

Marshfield, W. B. (1992). Submarine data set for use in autopilot research. Technical memorandum. DRA Haslar. DRA/MAR TM (MTH) 92314.

McPhail, S. D. and M. Pebody (1998). Navigation and control of an autonomous underwater vehicle using a distributed, networked, control architecture. *Journal of the Society for Underwater Technology* **23**(1), 19–30.

Menon, P. K. and V. R. Iragavarapu (1998). *Blended Homing Guidance Law Using Fuzzy Logic*. World Wide Web. Date Accessed, August 16th 2004. `http://www.optisyn.com/papers/1998/fuzzy_g.pdf`.

Millard, N. W., G. Griffiths, G. Finegan, S. D. McPhail, D. T. Meldrum, M. Pebody, J. R. Perrett, P. Stevenson and A. T. Webb (1998). Versatile autonomous submersibles — the realising and testing of a practical vehicle. *Journal of the Society for Underwater Technology* **23**(1), 7–17.

Morari, M. and J. M. Lee (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering* **23**, 667–682.

Naeem, W. (2002). Model predictive control of an autonomous underwater vehicle. In: *Proceedings UKACC Control 2002*. UKACC. Sheffield, UK. pp. 19–23.

Naeem, W., R. Sutton and J. Chudley (2003*a*). LQG/LTR control of an autonomous underwater vehicle using a hybrid guidance law. In: *Proceedings of Guidance and Control of Underwater Vehicles 2003*. Elsevier IFAC Publications. Newport, South Wales, UK. pp. 31–36.

Naeem, W., R. Sutton and J. Chudley (2003*b*). System identification, modelling and control of an autonomous underwater vehicle. In: *Maneouvring and Control of Marine Craft (MCMC2003)*. IFAC. Girona, Spain. pp. 37–42.

Naeem, W., R. Sutton and J. Chudley (2004*a*). Model predictive control of an autonomous underwater vehicle with a fuzzy objective function optimized using a GA. In: *Proceedings of Control Applications in Marine Systems (CAMS'04).* IFAC. Ancona, Italy. pp. 433–438.

Naeem, W., R. Sutton and S. M. Ahmad (2003*c*). Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking (invited paper). Presented in the World Maritime Technology Conference, October 17-20, San Francisco, California, USA, Paper B1.

Naeem, W., R. Sutton and S. M. Ahmad (2004*b*). Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking. *IMarEST Journal of Marine Science and Environment* C(1), 25–35.

Naeem, W., R. Sutton, J. Chudley, F. R. Dalgleish and S. Tetlow (2004*c*). A genetic algorithm-based model predictive control autopilot design and its implementation in an autonomous underwater vehicle. *IMechE Transactions Part M, Journal of Engineering for the Maritime Environment* **218**(M3), 175–188.

Naeem, W., R. Sutton, S. M. Ahmad and R. S. Burns (2003*d*). A review of guidance laws applicable to unmanned underwater vehicles. *Journal of Navigation* **56**(00), 1–15.

Oh, K. H., J.Y. Kim, I. W. Park, J. Lee and J. H. Oh (2002). A study on the control of AUV's homing and docking. In: *Proceedings of 9th IEEE Conference on Mechatronics and Machine Vision in Practice.* IEEE. Chiang Mai, Thailand. pp. 45–52.

Oliveira, P., C. Silvestre, A. Aguiar and A. Pascoal (1998). Guidance and control of the SIRENE underwater vehicle: from system design to tests at sea. In: *Proceedings Of Oceans'98.* Nice, France. pp. 1043–1048.

Qin, S. and T. A. Badgewell (2000). An overview of nonlinear model predictive control applications. In: *Nonlinear Model Predictive Control.* Switzerland.

Rajasekhar, V. and A. G. Sreenatha (2000). Fuzzy logic implementation of proportional navigation guidance. *Acta Astronautica* **46**(1), 17–24.

Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE Control Systems Magazine* pp. 38–52.

Richalet, J. (1993). Industrial applications of model based predictive control. *Automatica* **29**(5), 1251–1274.

Richalet, J., A. Rault, J. L. Testud and J. Papon (1978). Model predictive heuristic control: Applications to industrial processes. *Automatica* **14**, 413–428.

Rock, S. M., M. J. Lee, H. H. Wang, R. L. Marks and R. C. Burton (1992). Combined camera and vehicle tracking of underwater objects. In: *Proceedings of ROV'92.* San Diego, CA, USA. pp. 1–8.

Ruiz, I. Tena (2004). *Pipeline and Cable Tracking and Inspection AUV AUTOTRACKER.* World Wide Web. Date Accessed, August 11th 2004. `http://www.ece.eps.hw.ac.uk/~autotrack/`.

Sait, S. M. and H. Youssef (1999). *Iterative Computer Algorithms with Applications in Engineering, Solving Combinatorial Optimization Problems.* IEEE Computer Society.

Santina, M. S., A. R. Stubberud and G. H. Hostetter (1994). *Digital Control System Design.* 2nd ed.. Saunders College Publishing, Harcourt Brace College Publishers.

Seaeye, Marine (2004). *Falcon ROV Specifications.* World Wide Web. Date Accessed, August 11th 2004. `http://www.seaeye.com/falcon.html`.

Skogestad, S. and I. Postlethwaite (1996). *Multivariable Feedback Control: Analysis and Design using Frequency-domain Methods.* John Wiley and Sons Ltd.

Soeterboek, R. (1992). *Predictive Control, A Unified Approach.* Prentice Hall.

Sousa, J. M. and U. Kaymak (2001). Model predictive control using fuzzy decision functions. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics* **31**(1), 54–65.

Sutton, R., R. S. Burns and P. J. Craven (2000). Intelligent steering control of an autonomous underwater vehicle. *Journal of Navigation* **53**(3), 511–525.

Tabaii, S. S., F. El-Hawary and M. El-Hawary (1994). Hybrid adaptive control of autonomous underwater vehicle. In: *Proceedings of the 1994 IEEE Symposium on Autonomous Underwater Vehicle Technology.* IEEE Oceanic Engineering Society. Cambridge, MA, USA. pp. 275–282.

Tan, C. S., R. Sutton and J. Chudley (2003*a*). Collision avoidance systems for autonomous underwater vehicles, Part A: Obstacle detection. Submitted to IMarEST Journal of Marine Engineering and Technology, Part A.

Tan, C. S., R. Sutton and J. Chudley (2003*b*). Collision avoidance systems for autonomous underwater vehicles, Part B: Obstacle avoidance. Submitted to IMarEST Journal of Marine Engineering and Technology, Part A.

Tan, C. S., R. Sutton and J. Chudley (2004). An incremental stochastic motion planning technique for autonomous underwater vehicles. In: *Proceedings of Control Applications Marine Systems (CAMS'04).* IFAC. Ancona, Italy. pp. 483–488.

Taylor, A. (2000). *Oceanography, waves and wind.* World Wide Web. Date Accessed, July 27th 2004. `http://www.seafriends.org.nz/oceano/waves.htm`.

Tetlow, S. (2001). *The Offshore Technology Centre, Cranfield AUV.* World Wide Web. Date Accessed, July 27th 2004. `http://www.cranfield.ac.uk/sims/marine/research/oldauv.htm`.

Tinker, S. J., A. R. Bowman and T. B. Booth (1979). Identifying submersible dynamics from free model experiments. In: *RINA annual report and transactions.* pp. 191–196.

Tonge, A. (2000). Marlin, the UK military UUV programme - A programme overview. In: *Proceedings of International UUV Symposium2000.* Newport, RI, USA. pp. 31–38.

Triantafyllou, M. S. and M. A. Grosenbaugh (1991). Robust control for underwater vehicle systems with time delays. *IEEE Journal of Oceanic Engineering* **16**(1), 146–151.

Tsao, L. P. and C. S. Lin (2000). A new optimal guidance law for short-range homing missiles. *Proceedings of the National Science Council, Part A: Physical Science and Engineering, Republic of China* **24**(6), 422–426.

Wattenburg, F. (1996). *Spherical Coordinates and the GPS.* World Wide Web. Date Accessed, July 27th 2004. `http://www.math.montana.edu/frankw/ccp/cases/Global-Positioning/spherical-coordinates/learn.htm`.

Weerasooriya, S. and D. T. Phan (1995). Discrete-time LQG/LTR design and modelling of a disk drive actuator tracking servo system. *IEEE Transactions on Industrial Electronics* **42**(3), 240–247.

Wettergreen, D., C. Gaskett and A. Zelinsky (1999). Autonomous guidance and control for an underwater robotic vehicle. In: *International Conference on Field and Service Robotics, FSR'99.* Pittsburgh, USA. pp. 58–63.

Whitcomb, L. L. (2000). Underwater robotics: out of the research laboratory and into the field. In: *IEEE International Conference on Robotics and Automation.* USA.

Wilks, N. (2000). In-depth analysis. *Professional Engineering.*

Wolovich, W. A. (1994). *Automatic Control Systems, Basic Analysis and Design.* International Edition, Saunders College Publishing.

Yang, C. D. and H. Y. Chen (2001). Three-dimensional nonlinear H-infinity guidance law. *International Journal of Robust and Nonlinear Control* **11**(2), 109–129.

Yeo, D. J. (1999). *Design of AUV Tracking System using the Sliding Mode Control and the Optimal Control Theory.* Department of Naval Architecture and Ocean Engineering. MS Thesis, Seoul National University, Korea.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control* **8**, 338–353.

Zarchan, P. (1994). *Tactical and Strategic Missile Guidance.* 2nd ed.. AIAA, Inc., Washington DC, USA.

Appendix A

Hammerhead Electronics Schematic and Assembly Code

This appendix provides the schematic of a microcontroller based stepper motor controller card designed indigenously at the University of Plymouth. This circuit is currently being used in the *Hammerhead* AUV to control the onboard actuators. In addition, the assembly code based on Intel 8051 instruction set for the Atmel 89C2051 microcontroller used in the circuit is also given herein.

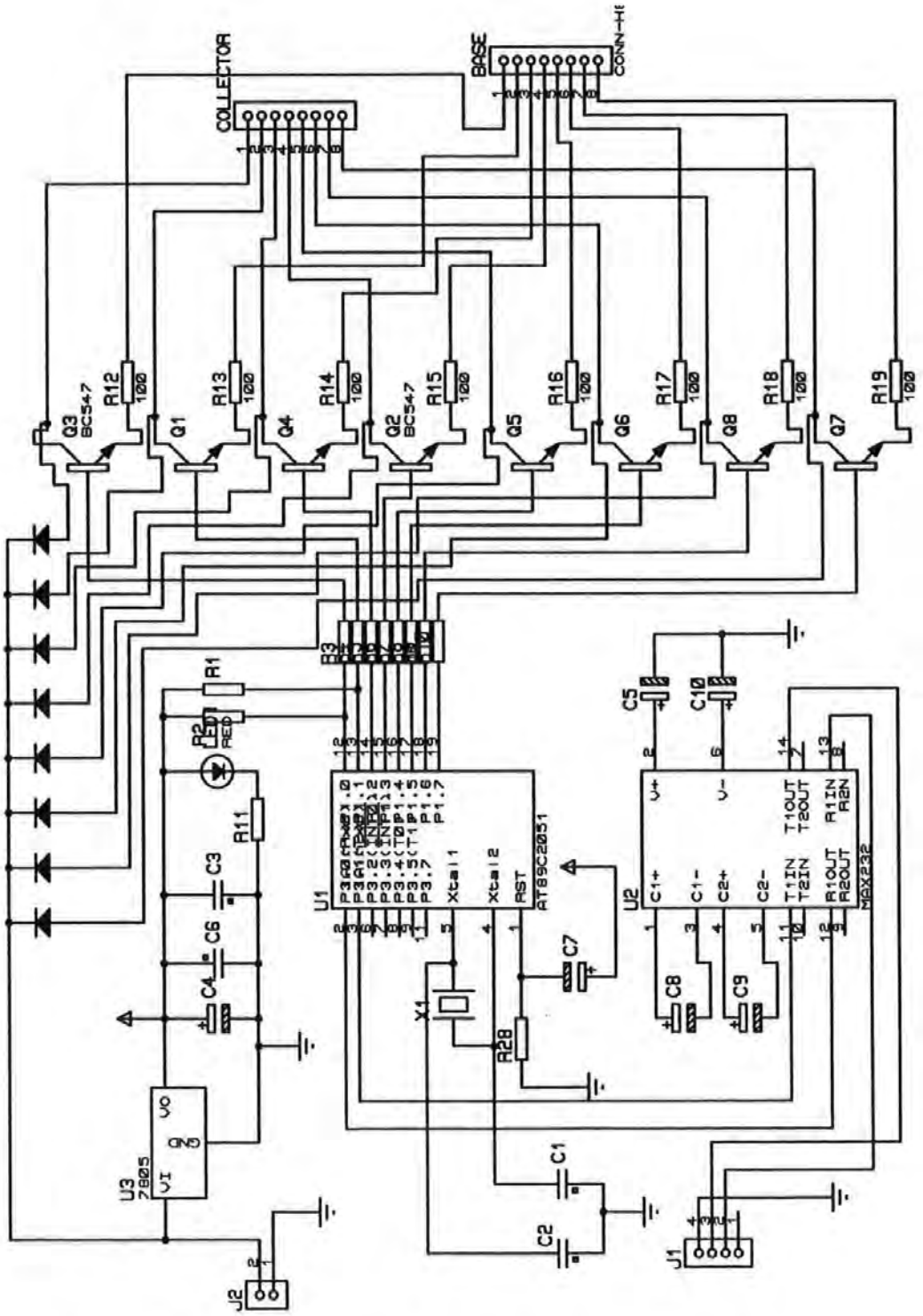


Figure A.1: Schematic of *Hammerhead* actuator controller card

A.1 Assembly Code for the Stepper Motor Controller

```
*************************************************************************
Stepper Motor Controller by Wasif Naeem 11/06/2002
based on 8051 Instruction Set
*************************************************************************
```

This program receive inputs from the serial port of the microcontroller connected to the serial port of the computer. The inputs transmitted to the controller consists of information on

- which stepper motor to turn
- whether to rotate clockwise or counterclockwise
- the number of pulses to be transmitted to the stepper motor

```
*************************************************************************
main program starts here

        ORG 0
        SJMP START
        ORG 20H
START:  MOV SCON,#52H
        MOV PCON,#0H
        MOV TH1,#0FDH ;set the baud rate to 9600
        MOV TMOD,#20H
        SETB TR1
        MOV P1,#0H

;*****************************************************************
;Receive the information byte containing the inputs mentioned above
;*****************************************************************

        SETB P3.0

MAIN:   CLR P3.7
        CLR P3.4
        CLR P3.5
RXBUF:  JNB RI,RXBUF
        CLR RI
        MOV A,SBUF ;received data from the serial buffer

;*****************************************************************
;Initialise counters and ports
;*****************************************************************
```

```
        MOV R3,#0H
        MOV R4,#0H
        MOV R0,#0H
        MOV P1,#0H ;2 cycles

;*********************************************************************
;Set Port 3 Bit 0 as an input port
;*********************************************************************

        MOV R4,A
        ANL A,#00111111B
        SETB P3.4

;*********************************************************************
;Check if the no.  of pulses are zero, in which case, terminate
;the program, otherwise continue
;*********************************************************************

        JNZ CONT
TEMP:   LJMP FINISH
CONT:   MOV A,R4
        POP A

;*********************************************************************
;Check if the command is for the rudder or hydroplane
;*********************************************************************

        SETB P3.5
        JB ACC.7,RUDDER
        JNB ACC.7,HYDRO

;*********************************************************************
;Check to see if the command is for Rudder clockwise
;or anticlockwise motion
;*********************************************************************

RUDDER: JB ACC.6,RCLOCK
        JNB ACC.6,RACLOCK

RCLOCK: CLR ACC.7 ;clear the mode bit in the accumulator
        CLR ACC.6
        MOV R0,A
        MOV P1,#0AH
        ACALL DELAYM
```

```
        MOV P1,#09H
        ACALL DELAYM
        MOV P1,#05H
        ACALL DELAYM
        MOV P1,#06H
        ACALL DELAYM
        DEC R0 ;1 cycle
        MOV A,R0
        JZ FINISH ;check if accumulator is zero
        SJMP RCLOCK ;2 cycles

RACLOCK:CLR ACC.7
        CLR ACC.6
        MOV R0,A
        MOV P1,#06H
        ACALL DELAYM
        MOV P1,#05H
        ACALL DELAYM
        MOV P1,#09H
        ACALL DELAYM
        MOV P1,#0AH
        ACALL DELAYM
        DEC R0 ;1 cycle
        MOV A,R0
        JZ FINISH ;check if accumulator is zero
        SJMP RACLOCK ;2 cycles

;*********************************************************************
;Check if the command is for Hydroplane clockwise
;or anticlockwise motion
;*********************************************************************

HYDRO:  SETB P3.7
        JB ACC.6,HCLOCK
        JNB ACC.6,HACLOCK

HCLOCK: CLR P3.2 ;for debugging purpose, normally high and should be
                 cleared
        CLR ACC.7 ;clear the mode bit in the accumulator
        CLR ACC.6
        MOV R0,A
        MOV P1,#0A0H
        ACALL DELAYM
        MOV P1,#90H
        ACALL DELAYM
```

```
        MOV P1,#50H
        ACALL DELAYM
        MOV P1,#60H
        ACALL DELAYM
        DEC R0 ;1 cycle
        MOV A,R0
        JZ FINISH ;check if accumulator is zero
        SJMP HCLOCK ;2 cycles

HACLOCK:CLR P3.3 ;for debugging purpose, normally high and should be
                 cleared
        CLR ACC.7
        CLR ACC.6
        MOV R0,A
        MOV P1,#60H
        ACALL DELAYM
        MOV P1,#50H
        ACALL DELAYM
        MOV P1,#90H
        ACALL DELAYM
        MOV P1,#0A0H
        ACALL DELAYM
        DEC R0 ;1 cycle
        MOV A,R0
        JZ FINISH ;check if accumulator is zero
        SJMP HACLOCK ;2 cycles

;*****************************************************************
;Delay in sending two consecutive pulses or Pulse duration
;*****************************************************************

DELAYM: MOV R2,#06H ;1 cycle 4
        ;(1+1+255*2+2+1+255*2+2+1+255*2+2+1+255*2+2)*1e-6 = 0.0021s
TAKEM1: MOV R1,#0FFH ;1 cycle FF
TAKEM:  DJNZ R1,TAKEM ;2 cycles
        DJNZ R2,TAKEM1 ;2 cycles
        RET ;2 cycles, 1*4*2*255*1e-6

FINISH: SETB P3.2
        SETB P3.3
        MOV SBUF,#11111111B

TXBUF:  JNB TI,TXBUF
        CLR TI
        CLR A
```

```
        MOV P1,#0H
        MOV R0,#0H
        LJMP MAIN

END
```

Appendix B

Stability Characteristics

In this appendix, several terms are defined which are vital for a frequency domain control system design. A Bode diagram is also shown which facilitates reader's understanding.

Two types of specifications are generally given prior to any controller design which are closely related. The time domain specifications involve the maximum overshoot, settling time etc. whereas the frequency domain specifications provide the bandwidth, gain margin, phase margin etc. of the system. These specifications can be evaluated by generating the step response and Bode plot of the system respectively. However, in an LQG design, frequency tuning is usually desired and therefore some definitions are given here for a clearer understanding. There are four main frequency domain parameters generally encountered in an LQG paradigm. Figure B.1 depicts these quantities and are defined below

Gain crossover frequency (gcf) is the frequency of the system at which the open loop gain of the plant is 0 dB on a Bode magnitude plot.

Phase crossover frequency (pcf) is the frequency at which the open loop phase of the plant reaches -180^0 on a Bode phase plot.

Gain margin The gain of the system at pcf below the 0 dB level is called the gain margin (GM). It is defined as the amount of gain in dB that can be added to the loop before the closed loop system becomes unstable. For stable systems, the GM is always positive and stays below the 0 dB line. Clearly, the robustness of the system to external disturbances depend on the magnitude of GM.

Phase margin The phase margin (PM) of the system can be found by first locating the gcf and measuring the distance between gcf on the phase plot and -180^0 line. This gap or distance is termed as the PM of the system as can be seen in Figure B.1. It can be defined as the amount of pure phase delay that can be added to the loop before the closed loop system becomes unstable.

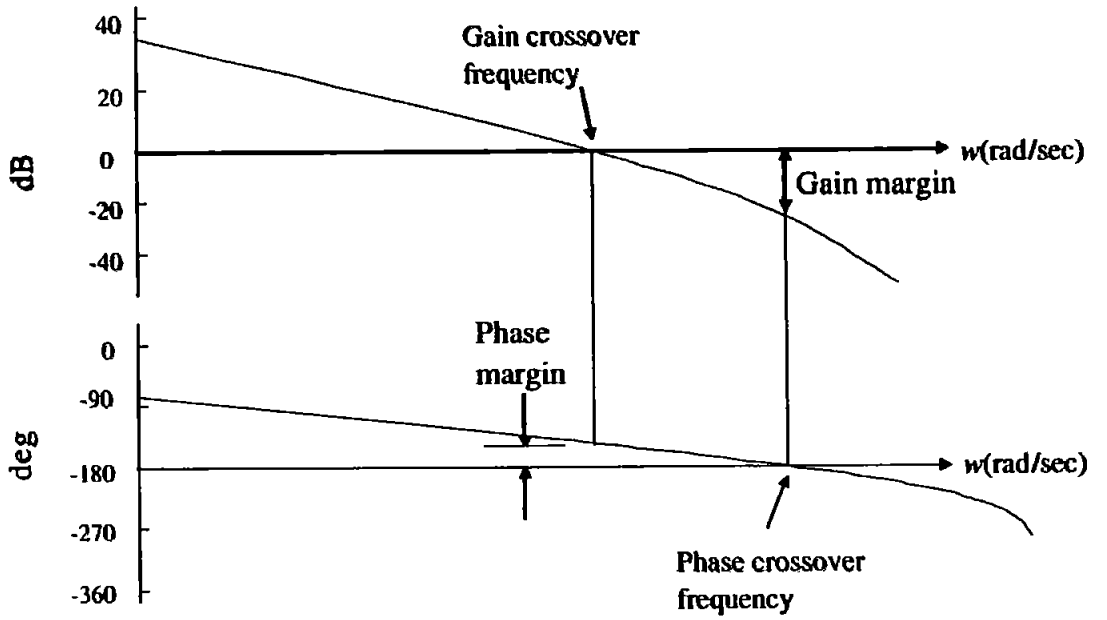


Figure B.1: Definition of stability margins on a Bode plot

Appendix C

Reference Input Tracking for a State Feedback Controller

A derivation has been carried out herein to evaluate the contents of blocks $\mathbf{N}_x$ and $\mathbf{N}_u$ in Figure 6.2 for reference input tracking in an LQR control strategy (Franklin *et al.*, 1998).

Let $\mathbf{N}_x$ denotes the forward block which transforms the reference input $\mathbf{r}$ to a reference state $\mathbf{x}_r$ that is an equilibrium one for that $\mathbf{r}$. Mathematically, this can be stated as

$$\mathbf{N}_x\mathbf{r} = \mathbf{x}_r \tag{C.1}$$

$$u = -\mathbf{K}_c(\mathbf{x} - \mathbf{x}_r) \tag{C.2}$$

The final or steady state value of the states can be written as

$$\mathbf{x}(\infty) = \mathbf{x}_{ss} = \mathbf{x}_r \tag{C.3}$$

Hence,

$$\mathbf{N}_x\mathbf{r} = \mathbf{x}_r = \mathbf{x}_{ss} \tag{C.4}$$

To compensate for any steady state output error in case of type 0 systems, a steady state control term is needed that is proportional to the reference input, i.e.,

$$u_{ss} = \mathbf{N}_u\mathbf{r} \tag{C.5}$$

Now, from Figure 6.2

$$\mathbf{C}_r\mathbf{x}_{ss} = y_r = \mathbf{r} \tag{C.6}$$

which reduces to

$$\begin{aligned} \mathbf{C}_r\mathbf{N}_x\mathbf{r} &= \mathbf{r} \\ \mathbf{C}_r\mathbf{N}_x &= \mathbf{I} \end{aligned} \tag{C.7}$$

Since the system is at steady state, i.e., $\mathbf{x}(k+1) = \mathbf{x}(k) = x_{ss}$ and $u = u_{ss}$, therefore

$$\begin{aligned} \mathbf{x}_{ss} &= \mathbf{A}\mathbf{x}_{ss} + \mathbf{B}u_{ss} \\ (\mathbf{A} - \mathbf{I})\mathbf{x}_{ss} + \mathbf{B}u_{ss} &= 0 \end{aligned}$$

substituting Equations C.5 and C.4 in the above equations

$$(\mathbf{A} - \mathbf{I})\mathbf{N}_x r + \mathbf{B}\mathbf{N}_u r = 0 \tag{C.8}$$

$$(\mathbf{A} - \mathbf{I})\mathbf{N}_x + \mathbf{B}\mathbf{N}_u = 0 \tag{C.9}$$

Finally writing Equations C.7 and C.9 in matrix form

$$\begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{B} \\ \mathbf{C}_r & 0 \end{bmatrix} \begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix}$$

and solving for $\mathbf{N}_x$ and $\mathbf{N}_u$ yields the desired result

$$\begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{B} \\ \mathbf{C}_r & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix} \tag{C.10}$$

Appendix D

Publications

The work within this thesis has contributed significantly to the underwater research literature via the following list of publications. This includes all the papers which have either been published, accepted or are under preparation.

Book Chapter(s)

- Loebis, D., **W. Naeem**, R. Sutton and J. Chudley (2004). The Navigation, Guidance and Control of the *Hammerhead* Autonomous Underwater Vehicle. To appear in Unmanned Marine Vehicles (Roberts, G. N. and Sutton, R. (Ed)). Peter Peregrinus Ltd., Herts.

Journal Papers

- **Naeem, W.**, R. Sutton, J. Chudley, F. R. Dalgleish and S. Tetlow (2004). A genetic algorithm-based model predictive control autopilot design and its implementation in an autonomous underwater vehicle. IMeChE Transactions Part M, Journal of Engineering for the Maritime Environment, pp. 175-188, vol. 218, no. M3, August.

- **Naeem, W.**, R. Sutton and S. M. Ahmad (2004). Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking. IMarEST Journal of Marine Science and Environment part C, no. 1, pp. 25-35, March.

- **Naeem, W.**, R. Sutton, S. M. Ahmad and R. S. Burns (2003). A review of guidance laws applicable to unmanned underwater vehicles. Journal of Navigation, vol. 56, no. 1, pp. 15-29, January.

- Loebis, D., R. Sutton, J. Chudley and **W. Naeem** (2004). Adaptive tuning of a Kalman filter via fuzzy logic for an intelligent AUV navigation system. Control Engineering Practice, vol. 12, no. 12, pp. 1531-1539.

Refereed Conference Papers

- **Naeem, W.**, R. Sutton and J. Chudley (2003). System identification, modelling and control of an autonomous underwater vehicle. In Proceedings of Manoeuvring and Control of Marine Craft (MCMC 2003), pp. 37-42, September, Girona, Spain

- **Naeem, W.**, R. Sutton and S. M. Ahmad (2003). LQG/LTR control of an autonomous underwater vehicle using a hybrid guidance law. In Proceedings of Guidance and Control of Underwater Vehicles (GCUV'03), pp. 31-36, April, Newport, UK

- **Naeem, W.**, R. Sutton and S. M. Ahmad (2003). Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking (Invited Paper). Presented in the World Maritime Technology Conference, Paper B1, October 17-20, San Francisco, California, USA

- **Naeem, W.** (2002). Model predictive control of an autonomous underwater vehicle. In Proceedings of United Kingdom Automatic Control Council (UKACC 2002) postgraduate symposium, pp. 19-23, September, Sheffield, UK

- **Naeem, W.**, R. Sutton and J. Chudley (2004). Model predictive control of an autonomous underwater vehicle with a fuzzy objective function optimized using a GA. In Proceedings of Control Applications in Marine Systems (CAMS'04), pp. 433-438, July, Ancona, Italy.

- Dalgleish, F. R., **W. Naeem**, S. Tetlow, R. L. Allwood and R. Sutton (2004). Adaptive AUV control for optimized swathe laser stripe imaging. In Proceedings of 3rd IFAC Symposium on Mechatronic Systems, pp. 175-180, September, Sydney, Australia.

- Tiano, A., R. Sutton, A. Lozowicki and **W. Naeem** (2004). Observer Kalman identification of an autonomous underwater vehicle. In Proceedings of Control Applications in Marine Systems (CAMS'04), pp. 269-272, July, Ancona, Italy.

- Loebis, D., R. Sutton, J. Chudley, **W. Naeem**, F. R. Dalgleish and S. Tetlow (2004). The application of soft computing techniques to an integrated navigation system of an AUV (Invited Paper). In Proceedings of 5th IFAC Symposium on Intelligent Autonomous Vehicles (IAV2004), Paper MA-3-2, July, Lisbon, Portugal.

NAVIGATION, GUIDANCE AND CONTROL OF THE *HAMMERHEAD* AUTONOMOUS UNDERWATER VEHICLE

D. Loebis, W. Naeem, R. Sutton and J. Chudley

d.loebis*, w.naeem, r.sutton, j.chudley@plymouth.ac.uk

MARINE AND INDUSTRIAL DYNAMIC ANALYSIS RESEARCH GROUP, SCHOOL OF ENGINEERING, THE UNIVERSITY OF PLYMOUTH, DRAKE CIRCUS, PLYMOUTH, PL4 8AA, UK

*corresponding author

Abstract: This chapter describes the navigation, guidance and control (NGC) of the *Hammerhead* autonomous underwater vehicle (AUV). The navigation system is based on an integrated use of the global positioning system (GPS) and several inertial navigation system (INS) sensors. A simple Kalman filter (SKF) and an extended Kalman filter (EKF) are proposed to be used subsequently to fuse the data from the INS sensors and to integrate them with the GPS data. The chapter highlights the use of fuzzy logic techniques optimized by multiobjective genetic algorithm (MOGA) to the adaptation of the initial statistical assumption of both the SKF and EKF caused by possible changes in sensor noise characteristics.

For controller design, the *Hammerhead* models are extracted using SI techniques on actual vehicle data obtained from full scale experiments. Two guidance laws are proposed which are designed for cable/pipeline inspection task for cruising type vehicles. The control systems developed for *Hammerhead* are a combination of optimal control and artificial intelligence (AI) strategies. A discrete time linear quadratic Gaussian controller with loop transfer recovery (LQG/LTR) is formulated. In addition, two forms of model predictive controllers (MPC) blended with a genetic algorithm (GA) and fuzzy logic are designed and tested in *Hammerhead*. Simulation as well as real-time results are presented.

Keywords: Autonomous underwater vehicle; Navigation; Sensor Fusion; Kalman filters; Extended Kalman filters; Fuzzy Logic; Genetic Algorithms; Guidance; System Identification; Optimal Control; Model Prediction Control

1 INTRODUCTION

The development of AUVs for scientific, military and commercial purposes in applications such as ocean surveying (Størkersen *et al.*, 1998), unexploded ordnance hunting (Wright *et al.*, 1996) and cable tracking and inspection (Asakawa *et al.*, 2000) requires the corresponding development of NGC systems, which should work in accord with each other for proper operation. Navigation systems are necessary to provide knowledge of vehicle position and attitude. The guidance systems manipulate the output of the navigation systems to generate suitable trajectories to be followed by the vehicle. This takes into account the target and any obstacles that may have been encountered during the course of a mission. The control systems are responsible for keeping the vehicle on course as specified by the guidance processor. In the *Hammerhead* AUV, this is achieved through manipulating the rudder and the hydroplane of the vehicle. The need for accuracy in NGC systems is paramount. Erroneous position and attitude data in navigation systems can lead to a meaningless interpretation of the collected data, which in turn affect the accuracy of the corresponding guidance and control systems. This, if not contained properly may lead to a catastrophic failure of an AUV during a specific mission. The integrated NGC of the *Hammerhead* AUV can be depicted in Figure 1.

Figure 1:

Hammerhead, shown in Figure 2, was developed from a deep mobile target (DMT) torpedo of $3(m)$ length and $30(cm)$ diameter that was purchased by Cranfield University (CU). Initial modifications were made to transform the torpedo into a PC controlled AUV (Naylies, 2000). Subsequently, research teams from the University of Plymouth (UP) and CU have successfully developed an integrated NGC system for the vehicle. In this collaborative work, CU developed

a navigation subsystem based on a laser stripe illumination methodology developed previously there (Tetlow and Allwood, 1995). Interested reader on the details of this subsystem is referred to Dalgleish (2004). The focus of this chapter is on the NGC work undertaken by the UP research team. The navigation system is based on a GPS/INS integrated system equipped with adaptive Kalman filtering (KF) techniques. The proposed guidance laws, namely the pure pursuit and hybrid guidance systems are formulated for cable/pipeline inspection task. Three approaches are suggested for autopilot design: linear quadratic Gaussian controller with loop transfer recovery (LQG/LTR), a model predictive controller (MPC) optimized using a GA and a fuzzy based GA-MPC.

Figure 2:

2 THE *HAMMERHEAD* AUV NAVIGATION SYSTEM

A growing number of research groups around the world are developing integrated navigation systems utilising INS and GPS (Gade and Jalving, 1999; Grenon *et al.*, 2001; Yun *et al.*, 1999). However, few of these works make explicit the essential need for fusion of several INS sensors that enable the users to maintain the accuracy or even to prevent a complete failure of this part of navigation system, before being integrated with the GPS. Kinsey and Whitcomb (2003), for example, use a switching mechanism to prevent a complete failure of the INS. Although simple to implement, the approach may not be appropriate to use to maintain a certain level of accuracy.

Several estimation methods have been used in the past for multisensor data fusion and integration purpose in AUVs (Loebis *et al.*, 2002). To this end, SKF/EKF and their variants have been popular methods in the past and interest in developing the algorithms has continued to the present day. However, a significant difficulty in designing a SKF/EKF can often be traced to incomplete a priori knowledge of the process covariance matrix ($\mathbf{Q}$) and measurement noise covariance matrix ($\mathbf{R}$). In most practical applications, these matrices are initially estimated or even unknown. The problem here is that the optimality of the estimation algorithm in the SKF/EKF setting is closely connected to the quality of a priori information about the process and measurement noise (Mehra, 1970). It has been shown that insufficiently known a priori filter statistics can reduce the precision of the estimated filter states or introduces biases to their estimates. In addition, incorrect a priori information can lead to practical divergence of the filter (Fitzgerald, 1971). From the aforementioned it may be argued that the conventional SKF/EKF with fixed ($\mathbf{Q}$) and/or ($\mathbf{R}$) should be replaced by an adaptive estimation formulation as discussed in the next section.

2.1 Fuzzy Kalman Filter

In this section, an on-line innovation-based adaptive scheme of the KF to adjust the $\mathbf{R}$ matrix employing the principles of fuzzy logic is presented. The fuzzy logic is chosen mainly because of its simplicity and closeness to human reasoning. These enable a satisfactory performance being developed empirically in practice without complicated mathematics. These have motivated the interest in the topic, as testified by related articles which have been appearing in the literature (Escamilla-Ambrosio and Mort, 2001; Jetto *et al.*, 1999; Kobayashi *et al.*, 1998).

The fuzzy logic Kalman filter (FKF) proposed herein is based on an innovation adaptive estimation (IAE) approach using a technique known as covariance-matching (Mehra, 1970). The basic idea behind the technique is to make the actual value of the covariance of the innovation sequences match its theoretical value.

The actual covariance is defined as an approximation of the Inn_k sample covariance through averaging inside a moving estimation window of size M (Mohamed and Schwarz, 1999) which takes the following form:

$$\hat{\mathbf{C}}_{Inn_k} = \frac{1}{M} \sum_{j=j_0}^{k} Inn_k \cdot Inn_k^T \tag{1}$$

where $j_0 = k - M + 1$ is the first sample inside the estimation window. An empirical heuristic experiment is conducted to choose the window size M that is adequate to capture the dynamic of the Inn_k actual covariance. From experimentation it was found that a good size for the moving window in Equation (1) used in this work is 15. The value of M is dependent on the dynamic of the Inn_k and therefore can vary for different types of applications.

The theoretical covariance of the innovation sequence is defined as (Mehra, 1970):

$$\mathbf{S}_k = \mathbf{H}_k \cdot \mathbf{P}_k^- \cdot \mathbf{H}_k^T + \mathbf{R}_k \tag{2}$$

The logic of the adaptation algorithm using covariance matching technique can be qualitatively described as follows. If the actual covariance value $\hat{\mathbf{C}}_{Inn_k}$ is observed, whose value is within the range predicted by theory $\mathbf{S}_k$ and the difference is very near to zero, this indicates that both covariances match almost perfectly and only a small change is needed to be made on the value of $\mathbf{R}$. If the actual covariance is greater than its theoretical value, the value of $\mathbf{R}$ should be decreased. On the contrary, if $\hat{\mathbf{C}}_{Inn_k}$ is less than $\mathbf{S}_k$, the value of $\mathbf{R}$ should be increased. This adjustment mechanism lends itself very well to being dealt with using a fuzzy-logic approach based on rules of the kind:

$$\textbf{IF } \langle \textit{antecedent} \rangle \textbf{ THEN } \langle \textit{consequent} \rangle \tag{3}$$

To implement the above covariance matching technique using the fuzzy logic approach, a new variable called $\mathbf{delta}_k$, is defined to detect the discrepancy between $\hat{\mathbf{C}}_{Inn_k}$ and $\mathbf{S}_k$. The following fuzzy rules of the kind of Equation (3) are used (Escamilla-Ambrosio and Mort, 2001):

$$\textbf{IF } \langle \mathbf{delta}_k \cong 0 \rangle \textbf{ THEN } \langle \mathbf{R}_k \textit{ is unchanged} \rangle \tag{4}$$

$$\textbf{IF } \langle \mathbf{delta}_k > 0 \rangle \textbf{ THEN } \langle \mathbf{R}_k \textit{ is decreased} \rangle \tag{5}$$

$$\textbf{IF } \langle \mathbf{delta}_k < 0 \rangle \textbf{ THEN } \langle \mathbf{R}_k \textit{ is increased} \rangle \tag{6}$$

Thus $\mathbf{R}$ is adjusted according to,

$$\mathbf{R}_k = \mathbf{R}_{k-1} + \Delta \mathbf{R}_k \tag{7}$$

where $\Delta \mathbf{R}_k$ is added or subtracted from $\mathbf{R}$ at each instant of time. Here $\mathbf{delta}_k$ is the input to the fuzzy inference system (FIS) and $\Delta \mathbf{R}_k$ is the output.

On the basis of the above adaptation hypothesis, the FIS can be implemented using three fuzzy sets for $\mathbf{delta}_k$; N = *Negative*, Z = *Zero* and P = *Positive*. For $\Delta \mathbf{R}_k$ the fuzzy sets are specified as I = *Increase*, M = *Maintain* and D = *Decrease*.

2.2 Fuzzy Logic Observer

To monitor the performance of a FKF, another FIS called the fuzzy logic observer (FLO) (Escamilla-Ambrosio and Mort, 2001) is used. The FLO assigns a weight or degree of confidence denoted as c_k, a number on the interval [0,1], to the FKF state estimate. The FLO is implemented using two inputs: the values of $|\mathbf{delta}_k|$ and $\mathbf{R}_k$. The fuzzy labels for the membership functions: Z = *Zero*, S = *Small* and L = *Large*. Three fuzzy singletons are defined for the output c_k and are labelled as G = *Good*, AV = *Average* and P = *Poor* with values 1, 0.5 and 0 respectively. The basic heuristic hypothesis for the FLO is as follows: if the value of $|\mathbf{delta}_k|$ is near to zero and the value of $\mathbf{R}_k$ is near to zero, then the FKF works almost perfectly and the state estimate of the FKF is assigned a weight near 1. On the contrary if one or both of these values increases far from zero, it means that the FKF performance is degrading and the FLO assigns a weight near 0. Table 1 gives the complete fuzzy rule base of each FLO.

Table 1:

2.3 Fuzzy Membership Functions Optimization

GAs in single- and multiobjective mode are used here to optimize the membership functions of the FKF. To translate the FKF membership functions to a representation useful as genetic material, they are parameterised with real-valued variables. Each of these variables constitutes a gene of the chromosomes for the MOGA. Boundaries of chromosomes are required for the creation of chromosomes in the right limits so that the MOGA is not misled to some other area of search space. The technique adopted in this thesis is to define the boundaries of the output membership functions according to the furthest points and the crossover points of two adjacent membership functions. In other words, the boundaries of FKF consist of three real-valued chromosomes (*Ch*s), as in Figure 3.

Figure 3:

The trapezoidal membership functions' two furthest points, -0.135 (D_1), -0.135 (D_2) and 0.135 (I_3), 0.135 (I_4) of FKF, remain the same in the GA's description to allow a similar representation as the fuzzy system's definition. As can be seen from Figure 3, D_3 and M_1 can change value in the 1^{st} *Ch* boundary, D_4, M_2 and I_1 in the 2^{nd} *Ch* boundary, and finally, M_3 and I_2 in 3^{rd} *Ch*. Table 2 shows the encoding used for optimization of the membership functions.

Table 2:

2.4 Implementation Results

This section discusses the implementation of the FKF optimized using MOGA discussed earlier for fusing heading data acquired during a real-time experiment conducted in Roadford Reservoir. The *Hammerhead* AUV second order model used herein was derived using SI techniques (Ljung, 1999), which will be discussed in Section 3. In this model, yaw (x_1) and delayed yaw (x_2) are the states of the system. It is assumed in this model that the forward velocity of the vehicle is constant at $1(m/sec)$ and the vehicle is not at an angle of roll and pitch. Process and measurement noise components are both zero mean white noise. Input to the system (indicated by δ_{r_k}) is rudder deflection. This model is assumed to be sufficiently accurate to represent the dynamics of the vehicle, and for this reason, any output produced by the model after being excited by an input, can be considered as an actual output value. This assumption also motivates the use of the model output as a reference in measuring the performance of the FKF-MOGA algorithm.

To test the FKF-MOGA algorithms, real data obtained from a TCM2 electronic compass and an inertial measurement unit (IMU), as a response to the input shown in Figure 4(a), are fused together with two sets of simulated data. To produce the simulated data, the noise in Figure 5(a) and 5(b) are simply added to the TCM2 electronic compass and IMU real data respectively. In this particular scenario, the second TCM2 electronic compass (sensor-3) is located in close proximity to the propeller DC motor of the vehicle, whose internal temperature increases with time and affects the sensor ambient temperature. A similar scenario can also be considered to occur when the second IMU (sensor-4) is located in close proximity to the laser unit used in the VNS whose initial internal temperature is high and settles down after sometime. This particular scenario can result in the noise characteristic shown in Figure 5(b).

Figure 4:
Figure 5:

The initial condition are:

$$x_0 = \begin{bmatrix} 0(rad) \\ 0(rad) \end{bmatrix}; \mathbf{P}_0 = \begin{bmatrix} 0.01(rad)^2 & 0 \\ 0 & 0.01(rad)^2 \end{bmatrix} \tag{8}$$

and $\mathbf{Q}_k$ is made constant as

$$\mathbf{Q}_k = \begin{bmatrix} 0(rad)^2 & 0 \\ 0 & 0.1725 \times 10^{-7}(rad)^2 \end{bmatrix} \tag{9}$$

The values of $\mathbf{P}_0$ and $\mathbf{Q}_k$ are determined heuristically. In real-time applications, the $\mathbf{Q}_k$ values are dependent on temporal and spatial variations in the environment such as sea conditions, ocean current, and local magnetic variations and therefore, appropriate adjustments to the initial values of $\mathbf{Q}$ also need to be undertaken. However, given the fact that the *Hammerhead* AUV mostly operates in a stable environment, the problem with the $\mathbf{Q}$ adjustment is reserved for future work. The actual value of $\mathbf{R}$ is assumed unknown, but its initial value is selected according to the heading accuracy of the sensors, *i.e.* $1(deg)^2$.

The covariance matching technique discussed previously is then implemented to maintain the performance of the estimation process. Subsequent optimization of $\Delta \mathbf{R}_k$ membership functions

using MOGA is done using the parameters shown in Table 3. Trade-off graphs of this particular search is shown in Figure 6.

Table 3:
Figure 6:

Table 4 shows the performance of the sensors, indicated by J_{zv} and J_{ze}, where

$$J_{zv} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(za_k - z_k)^2} \tag{10}$$

$$J_{ze} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(za_k - \hat{z}_k)^2} \tag{11}$$

Here, za_k is the actual value of the yaw, z_k is the measured yaw, $\hat{z}_k$ is the estimated yaw at an instant of time k and n = number of samples. A close look on the J_{zv} and J_{ze}, of each sensor indicates that the FKF with GA (single objective optimization) has improved the accuracy of the heading information of sensor-1 to sensor-4. However, the result of fusing the estimated sensor data has shown a slightly inferior performance, indicated by $J_{ze} = 0.2487(rad)$, compared to the performance of sensor-1, indicated by $J_{zv} = 0.2340(rad)$. This can be understood as a direct result of fusing a relatively accurate sensor-1, with other sensors that are less accurate. A further comparison is made between individual sensor performance of GA and MOGA case. It is clear that the individual sensor performance of the MOGA case, with the exception of sensor-1, has produced some improvements, with sensor-2 as the most noticeable one. It is clear that the improvement on sensor-2 has brought about an overall significant improvement on the quality of the estimation of the MOGA fused sensor, which is indicated by $J_{ze} = 0.2088(rad)$.

2.5 GPS/INS Navigation

Here, the fused estimated yaw obtained previously is treated as a single imaginary yaw sensor and used by other INS sensors to transform data from a body co-ordinate to a geographical (North-East-Down/NED) co-ordinate frame where integration with converted GPS data is performed using a combination of FKF and EKF techniques and can be referred to as fuzzy extended Kalman filter(FEKF). Two GPS/INS scenarios are considered. The first scenario is where the vehicle performs a surface mission. The second scenario is where the vehicle performs an underwater mission.

2.5.1 2D/Surface Mission

A continuous time model of the vehicle motion appropriate to this problem is taken to be

$$\dot{X}(t) = \mathbf{F}(X(t)) + W(t) \tag{12}$$

$$Z(t) = \mathbf{H}(X(t)) + V(t) \tag{13}$$

Denoted by $X(t) = [\ X_{NED}(t)\ \ Y_{NED}(t)\ \ \psi_{im}(t)\ \ r(t)\ \ u(t)\ \ v(t)\ ]^T$ are the model states. $X_{NED}(t)$ and $Y_{NED}(t)$ are the longitude and latitude of the AUV position converted from deg-min-sec in an Earth-centered Earth-fixed co-ordinate frame into metres in the NED co-ordinate frame, $\psi_{im}(t)$ is the yaw angle obtained from the imaginary yaw sensor, $r(t)$ is yaw rate, $u(t)$ and $v(t)$ are the surge and sway velocity respectively. In this system model, $\mathbf{F}(\cdot)$ and $\mathbf{H}(\cdot)$ are both continuous function, continuously differentiable in $X(t)$. The $W(t)$ and $V(t)$ are both zero mean white noise for the system and measurement models respectively. The model states are related through the following kinematically based set of functions ($\mathbf{F}(X(t))$) in Equation (12):

$$\dot{u}(t) = 0 \tag{14}$$

$$\dot{v}(t) = 0 \tag{15}$$

$$\dot{\psi}_{im}(t) = r(t) \tag{16}$$

$$\dot{r}(t) = 0 \tag{17}$$

$$\dot{X}_{NED}(t) = u(t)\cos\psi_{im}(t) - v(t)\sin\psi_{im}(t) \tag{18}$$

$$\dot{Y}_{NED}(t) = u(t)\sin\psi_{im}(t) + v(t)\cos\psi_{im}(t) \tag{19}$$

To obtain an EKF with an effective state prediction equation in a simple form, the continuous time model of Equations (14) - (19) have been linearised about the current state estimates, producing:

$$\mathbf{F}_{2D-linearised}(t) = \begin{bmatrix} 0 & 0 & -u(t)\sin\psi_{im}(t) - v(t)\cos\psi_{im}(t) & 0 & \cos\psi_{im}(t) & -\sin\psi_{im}(t) \\ 0 & 0 & u(t)\cos\psi_{im}(t) - v(t)\sin\psi_{im}(t) & 0 & \sin\psi_{im}(t) & \cos\psi_{im}(t) \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{20}$$

The output measurements are related through the states by the following matrix:

$$\mathbf{H}_{2D-linearised} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{21}$$

when GPS signal is available, and when it is not,

$$\mathbf{H}_{2D-linearised} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{22}$$

where $\mathbf{F}_{2D-linearised}$ and $\mathbf{H}_{2D-linearised}$ are respectively, equivalent to $\mathbf{A}$ and $\mathbf{C}$ in linear dynamic system. Subsequent discretisation with period $\mathrm{T} = 0.125(sec)$ of the linearised model results in the EKF algorithm. The initial conditions are:

The initial conditions are $X_0 = 0\mathbf{I}_{6\times 6}$ and $\mathbf{P}_0 = 0.01\mathbf{I}_{6\times 6}$, and $\mathbf{Q}$ is made constant as

$$\begin{bmatrix} 10(m)^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10(m)^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0175(rad)^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1(rad/sec)^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1(m/sec)^2 & \\ 0 & 0 & 0 & 0 & 0 & 0.1(m/sec)^2 \end{bmatrix} \quad (23)$$

The actual value of $\mathbf{R}$ is assumed unknown but its initial value is selected as:

$$\begin{bmatrix} 1000(m)^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1000(m)^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0873(rad)^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0175(rad/sec)^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(m/sec)^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(m/sec)^2 \end{bmatrix} \quad (24)$$

The FEKF algorithm is then implemented to the diagonal element of $\mathbf{R}_k$.

Figure 7:

Figure 7(a) shows the *Hammerhead* AUV trajectory obtained using GPS, dead reckoning using INS sensors (through double integration of the accelerometer data with respect to time) and integrated GPS/INS. As the initial value of $\mathbf{R}$ for both $X_{NED}(t)$ and $Y_{NED}(t)$ is $1000(m^2)$, the standard EKF algorithm puts less weight on the position obtained by GPS and more on the prediction of position obtained from dead reckoning method (using INS sensor data). Figure 7(b) shows that the matrix has been adjusted accordingly and more weight is given to the GPS data, and therefore the estimated trajectory in the integrated GPS/INS is "pulled" a little bit further to the GPS trajectory. However, big discrepancies can still be appreciated between the integrated GPS/INS estimate with respect to the GPS fixes. There are several explanations to this erratic behaviour. The first possibility is that it is caused by the poor level of accuracy of the low-cost GPS being used in this particular application. It is important to note that the proposed algorithm has detected a persistent high actual covariance ($\hat{\mathbf{C}}_{Inn_k}$) for both X_{NED} and Y_{NED} throughout the trajectory. This results in insufficient weight being given to the GPS fixes in the FEKF and more on the position obtained by the dead reckoning. The second possibility is that the GPS receiver did not lock into a sufficient number of satellites with a sufficiently small value of position dilution of precision (PDOP) that can provide the required level of accuracy. The use of a differential global positioning system (DGPS) receiver or a GPS receiver with a wide area augmentation system (WAAS) or a European geostationary navigation overlay service (EGNOS) capability can be considered as a way forward to alleviate this problem.

2.5.2 3D/Surface-Depth Mission

Many missions performed by AUVs require the vehicle to operate not only on the surface of the sea, but also at a particular depth. Examples of such AUVs and their specific missions can be found in Loebis *et al.* (2002). The *Hammerhead* AUV is also designed to be able to dive to a certain depth and perform a particular mission, such as tracking underwater cables for maintenance purposes or landmark recognition for an underwater absolute positioning system as proposed in Loebis *et al.* (2003). To carry out these missions, the *Hammerhead* AUV is equipped with underwater image acquisition techniques (Dalgleish *et al.*, 2003), coupled with a laser stripe illumination methodology (LSI) developed previously by Cranfield University (Tetlow

and Allwood, 1995) to provide an enhanced viewing of the seabed below the vehicle, and a depth controller developed by Naeem (2004), which will be discussed more thoroughly in the section to come about guidance and control.

The concept of 3D navigation system enhanced by the proposed techniques is demonstrated in this section. The real data used herein are those generated by the individual TCM2 electronic compass and IMU, their respective simulated counterparts, and their overall fused values. Further real-time experiments are considered to be imperative and must be conducted before a full-scale pseudo real-time implementation of the proposed techniques can be undertaken. This, however, due to the amount of time required to do so and to analyse the data produced thereby, is considered to be suitable for the future work of the *Hammerhead* AUV.

The mission scenario adopted in this section is designed to mimic the actual cable-tracking or landmark recognition that will be performed in the future by the *Hammerhead* vehicle. This involves acquiring GPS/INS data on the surface and subsequently finding the estimated trajectory before sending the vehicle to a certain depth. Once the vehicle is under the water, the GPS signals are completely blocked and the GPS/INS navigation system is replaced by a pure dead reckoning navigation system. During this period, the underwater image acquisition algorithms continuously observing the area beneath the vehicle to find a cable to be tracked or underwater landmarks to be identified and used as underwater absolute position fixes. In conditions where sufficient illumination is available in identifying those objects, produced either by the LSI or natural ambient light, the vehicle is then controlled to maintain its current depth. Otherwise, the depth controller algorithm will act accordingly and send the vehicle further down until sufficient illumination is obtained. After a certain period of time the vehicle is sent back to the surface to obtain GPS fixes that are used to reset the drift or the accumulated error produced by the dead reckoning navigation system.

The surge, sway and heave of the vehicle are obtained by integrating body co-ordinate frame acceleration data. The true values of the surge and sway are, respectively, defined as $1.3(m/sec)$ and $\pm 0.1(m/sec)$. The heave values are defined into five parts. The first is the heave of the vehicle when it is operating on the surface, *i.e.*, true values are assumed to be $0(m/sec)$. The second part is the heave of the vehicle as it is descending to a certain depth, defined here as $-0.1(m/sec)$. Once the vehicle reaches this, the depth controller is employed to maintain the depth of the vehicle. Consequently, the true heave during this period is defined to be $0(m/sec)$. The vehicle is sent back to the surface, and the heave during ascending period is defined to be $0.1(m/sec)$. Finally, the vehicle is back to the surface and the heave once again defined to be $0(m/sec)$. It clear that the errors added to these true values will contribute to the total drift suffered by the dead reckoning navigation system in finding the position of the vehicle when it is operating under the water.

$X(t)=[\ X_{NED}(t)\ \ Y_{NED}(t)\ \ Z_{NED}(t)\ \ \theta(t)\ \ q(t)\ \ \psi_{im}(t)\ \ r(t)\ \ u(t)\ \ v(t)\ \ w(t)\ ]^T$ is the state vector for a continuous time model of the vehicle motion appropriate to this problem, which is taken to be as in Equations (12) and (13), with $X_{NED}(t)$ and $Y_{NED}(t)$ are the longitude and latitude of the AUV position converted from deg-min-sec in the Earth-centered Earth-fixed co-ordinate frame into metres in the NED co-ordinate frame, $Z_{NED}(t)$ is the depth of the vehicle, $\theta(t)$ is the pitch, $q(t)$ is the pitch rate, $\psi_{im}(t)$ is the yaw angle obtained from the imaginary yaw sensor, $r(t)$ is yaw rate, $u(t)$, $v(t)$ and $w(t)$ are the surge, sway and heave velocity respectively.

The model states are related through the following kinematically based set of functions:

$$\dot{X}_{NED}(t) = u(t)\cos\psi_{im}(t)\cos\theta(t) - v(t)\sin\psi_{im}(t) + w(t)\cos\psi_{im}(t)\sin\theta(t) \quad (25)$$

$$\dot{Y}_{NED}(t) = u(t)\sin\psi_{im}(t)\cos\theta(t) + v(t)\cos\psi_{im}(t) + w(t)\sin\psi_{im}(t)\sin\theta(t) \quad (26)$$

$$\dot{Z}_{NED}(t) = -u(t)\sin\theta(t) + w(t)\cos\theta(t) \quad (27)$$

$$\dot{\theta}(t) = q(t) \quad (28)$$

$$\dot{q}(t) = 0 \quad (29)$$

$$\dot{\psi}_{im}(t) = r(t) \quad (30)$$

$$\dot{r}(t) = 0 \quad (31)$$

$$\dot{u}(t) = a_{x_{BODY}\rightarrow NED}(t) \quad (32)$$

$$\dot{v}(t) = a_{y_{BODY}\rightarrow NED}(t) \quad (33)$$

$$\dot{w}(t) = a_{z_{BODY}\rightarrow NED}(t) \quad (34)$$

where $a_{x_{BODY}\rightarrow NED}(t)$, $a_{y_{BODY}\rightarrow NED}(t)$ and $a_{z_{BODY}\rightarrow NED}(t)$ are the acceleration of the vehicle acquired in the body co-ordinate frame and transformed subsequently to the NED co-ordinate frame.

The output measurements are related through the states by an identity matrix $\mathbf{I}_{10\times10}$ when the vehicle is operating on the surface. When the vehicle is operating under the water, pure dead reckoning is used. Linearisation about the current estimates of the continuous time model of Equations (25)-(34), producing the $\mathbf{F}(t)$ of the system, which in this case defined as $\mathbf{F}_{3D-linearised}(t)$, with the corresponding $\mathbf{H}_{3D-linearised}(t) = \mathbf{I}_{10\times10}$. The FEKF algorithm is then implemented after subsequent discretisation with period T = 0.125(sec). The initial conditions are $X_0 = 0\mathbf{I}_{10\times10}$ and $\mathbf{P}_0 = 0.01\mathbf{I}_{10\times10}$, and $\mathbf{Q}$ is made constant as with the following components:

$\mathbf{Q}_{[1,1]} = 0.01(m)^2$
$\mathbf{Q}_{[2,2]} = 0.01(m)^2$
$\mathbf{Q}_{[3,3]} = 0.01(m)^2$
$\mathbf{Q}_{[4,4]} = 0.000001(rad)^2$
$\mathbf{Q}_{[5,5]} = 0.01(rad/sec)^2$
$\mathbf{Q}_{[6,6]} = 0.000001(rad)^2$
$\mathbf{Q}_{[7,7]} = 0.01(rad/sec)^2$
$\mathbf{Q}_{[8,8]} = 0.01(m/sec)^2$
$\mathbf{Q}_{[9,9]} = 0.01(m/sec)^2$
$\mathbf{Q}_{[10,10]} = 0.01(m/sec)^2$

The initial value of $\mathbf{R}$ is selected as:

$\mathbf{R}_{[1,1]} = 10(m)^2$
$\mathbf{R}_{[2,2]} = 10(m)^2$
$\mathbf{R}_{[3,3]} = 5(m)^2$
$\mathbf{R}_{[4,4]} = 0.000001(rad)^2$
$\mathbf{R}_{[5,5]} = 0.000001(rad/sec)^2$
$\mathbf{R}_{[6,6]} = 0(rad)^2$

$\mathbf{R}_{[7,7]} = 0(rad/sec)^2$
$\mathbf{R}_{[8,8]} = 2(m/sec)^2$
$\mathbf{R}_{[9,9]} = 2(m/sec)^2$
$\mathbf{R}_{[10,10]} = 2(m/sec)^2$

Figure 8 and 9 show the result of implementing the proposed FEKF algorithm to the 3D/surface-depth mission described herein using the yaw produced by an individual sensor (here represented by sensor-4) and fused sensor respectively. Readers interested on the complete results and analysis are referred to Loebis (2004).

Figure 8:

It is clear from Figure 8, that the initial GPS/INS surface trajectory using the yaw produced by sensor-4 contains an unexpected drift in vertical direction. This is a direct result of assuming the measurement noise in this direction as being higher than its corresponding process noise. The values of the measurement and process covariance matrices are indicated respectively as $\mathbf{R}_{[3,3]} = 5(m)^2$ and $\mathbf{Q}_{[3,3]} = 0.01(m)^2$. Consequently, the EKF algorithm puts more confidence on the process, *i.e.*, integrating the $\dot{Z}_{[NED]}$, than the measurement of depth from the pressure transducer. It is clear, as indicated by Equation (27), that integrating the value of $\dot{Z}_{[NED]}$ consequently integrates the noise in the $u(t)$, ψ_{im}, w and θ. This in turn produces an accumulation of error and needs to be reset to $Z_{NED} = 0(m)$, right before the vehicle dives. Once the vehicle is below the surface, the depth controller and the underwater image acquisition algorithms will work side by side to find objects of interest and to maintain a constant depth thereafter for a specific period of time. The vehicle is then sent back to the surface to obtain GPS fixes used to reset the drift produced by the dead reckoning process during the underwater mission. A similar case of DR error also occurs at this stage. Although the depth has been reset to $0(m)$, the EKF algorithm soon puts more confidence on the vertical DR process and consequently produces an estimate of depth larger than $0(m)$. This also happens to the horizontal (X_{NED} and Y_{NED}) estimation process. As the measurement covariance matrices for both the longitude and latitude are $\mathbf{R}_{[1,1]} = \mathbf{R}_{[2,2]} = 10(m)^2$, the estimation process put more weight on the DR processes, which are assumed to have $\mathbf{Q}_{[1,1]} = \mathbf{Q}_{[2,2]} = 0.01(m)^2$ process covariance matrices.

Figure 9:

Figure 9 shows the trajectory produced using the fused yaw sensor and with the values of $\mathbf{R}$ adjusted by FEKF. It is clear as presented, that as the assumed values of $\mathbf{R}$ for the longitude and latitude quite low, $10(m)^2$, compared to the true ones, which are simulated to be $225(m)^2$ and $100(m)^2$, the FEKF estimation process put initial weight more on the GPS fixes than the dead reckoning solutions. However, as the filter learns the true nature of $\mathbf{R}$ of these quantities, the FEKF makes an appropriate adjustment by putting more weight on the dead reckoning solution than on the GPS fixes. It can also be observed how the filter learns the true value of $\mathbf{R}_{[3,3]}$, which is simulated to be $0.0001(m)^2$. This time the vehicle is not estimated to have depth larger than $0(m)^2$, as in the case with the trajectory using yaw data produced by sensor-4. As before, once the vehicle is below the surface, the depth controller and the underwater image acquisition algorithms will work side by side to find objects of interest and to maintain a constant depth thereafter for a specific period of time. The vehicle is then sent back to the surface to obtain GPS fixes used to reset the drift produced by the dead reckoning process during the underwater mission. It is also clear here how the FEKF has learned the true nature of the $\mathbf{R}$ values. It can be observed from Figure 9 how the FEKF algorithm puts extra confidence on the GPS fixes

right after the vehicle reaches the surface. Soon afterwards however, the algorithm recognises the high level of noise inherent in the acquired GPS signals and put less confidence thereon. Small discrepancies still exist between the true end and the estimated end of the mission. However, it is clear that without the FEKF, the estimated end could easily coincide with the last GPS fix and cause a significant position error.

3 SYSTEM MODELLING

All controller designs are based on a model of the physical system to be controlled. This gives the modelling process utmost importance before any real time controller can be developed. It is imperative that the designer gain significant depth into system behaviour via extensive simulations using a model of the process as an alternative to the physical system. Clearly, this requires a model that can replicate the systems dynamic behaviour as closely as possible.

Modelling an underwater vehicle is a complex task because of the nonlinear nature of the vehicle dynamics and the degrees of freedom of vehicle movement. In addition, cross coupling effects makes the controller design even more intricated. Fortunately, there is a plentiful amount of literature available on the mathematical modelling of underwater vehicles and is generally applicable to all types of underwater vessels. However, a major difficulty in using these generalised models is the evaluation of hydrodynamic coefficients which require tank tests on a full scale physical model of the vehicle provided the test facility is available.

An alternate route to modelling an AUV using SI is thus suggested and used herein. A SI approach is useful in providing reliable and accurate models in a short time without relying too much on mathematical modelling techniques. This feature therefore, is attractive for the underwater vehicle manufacturers, where a vehicle configuration changes frequently to suit the mission requirements.

AUV modelling using SI approaches have been investigated before, (Tinker *et al.*, 1979; Ippoliti *et al.*, 2001; Goheen and Jefferys, 1990; Ahmad and Sutton, 2003), but most of the work involved has been done on identifying a model by generating data from a mathematical model of the vehicle. However, for *Hammerhead*, the SI is performed on input output data obtained from actual in water experiments. SI theory is well established and the reader is refered to Ljung (1999) for a comprehensive treatment of the subject.

3.1 Identification Results

Trials for SI have been performed at South West Water's Roadford Reservoir, Devon, and at Willen Lake in Milton Keynes. Experiments were designed that could obtain the best possible data for model development. Ideally, the requirement is to have a completely noise free data which is impossible in a real world environment. The *Hammerhead* is a low speed AUV that swims at approximately $2(knots)$. This gives some insight about the sampling period to be chosen. Clearly, too high sampling rate in this case will give no advantage whatsoever. A sampling rate of $1(Hz)$ is thus chosen iteratively which is adequate to obtain ample dynamical information about the system. By the same token, the frequency for the input signal is chosen as $0.1(Hz)$ which was deemed sufficient to excite the interesting modes of the system.

Some common type of excitation signals used are the uniformly distributed random numbers, pseudo random binary sequence, and its variants such as multistep, and doublet input. For depth channel identification, it was found that the vehicle would hit the bed without the depth autopilot in the loop for long duration inputs. Therefore, it was decided to perform several short duration experiments with a different multistep input being applied to the hydroplane in each experiment. The experiments can easily be merged for model identification using the SI toolbox in MATLAB. Please note that all available measurements were pre-filtered and resampled at $1(Hz)$ before any model parameters could be identified. Due to this, most of the high frequency contamination was eliminated and hence it was decided to extract an autoregressive with exogeneous (ARX) or state space model without modelling the noise separately.

3.1.1 Rudder-Yaw Channel

The input to this channel is the rudder deflections and the output is the vehicle's yaw or heading angle. The heading information is available from an onboard TCM2 compass and IMU. However, the vehicle response was obtained from TCM2 in these particular trials. It has been observed that that there is no strong cross coupling present between the yaw and depth channels. Moreover, the roll data remains unaltered with respect to the change in vehicle's heading. A single-input single-output (SISO) model has thus been developed for this channel from the data and is given by Equation (35).

$$G(q^{-1}) = \frac{-0.04226q^{-1} + 0.003435q^{-2}}{1 - 1.765q^{-1} + 0.765q^{-2}} \tag{35}$$

where q^{-1} is the delay operator. This model has been verified by independent data sets that were not used in the modelling process. In addition, correlation based tests were also carried out to gauge the quality of the model which were found to be adequate.

3.1.2 Hydroplane-Depth Channel

The input to this channel is the hydroplane deflections whilst the output is the depth of vehicle taken from a pressure transducer. It has been mentioned that only multistep inputs were employed to excite the depth dynamics of the vehicle due to the reasons explained before. The vehicle was allowed to swim freely in 6 degrees of freedom, however, only a single input (hydroplane) was manipulated and the heading, depth, roll and pitch data were analysed and recorded.

The *Hammerhead* data obtained from the depth trials reveal some cross coupling effects between the depth and heading angle. A multivariable model is therefore should be the ideal choice. However, exploiting the fact that the heading angle does not vary significantly when the vehicle is fully submerged, a SISO model involving only hydroplane deflections and depth has been developed. Several data sets containing these parameters were collected and suitable data were averaged, resampled and then merged to estimate the model coefficients.

A fourth order ARX model was chosen iteratively which gives the best fit between measured and model predicted outputs. Equation (36) below presents the ARX(441) model of the depth dynamics of *Hammerhead*

$$G(q^{-1}) = \frac{0.002681q^{-1} - 0.00327q^{-2} - 0.0007087q^{-3} + 0.001322q^{-4}}{1 - 3.6773q^{-1} + 5.0839q^{-2} - 3.1348q^{-3} + 0.72826q^{-4}} \tag{36}$$

The depth dynamic model has also been verified by analysing the correlation tests and time domain cross validation.

4 GUIDANCE

For *Hammerhead*, two novel guidance strategies have been proposed. These guidance laws were specifically designed for underwater cable inspection tasks for cruising type vehicles. The first guidance strategy presented has been borrowed from airborne systems and is termed as pure pursuit guidance. This scheme generate command signals which are proportional to the line of sight (LOS) angle so that the pursuing vehicle maintains its flight profile aligned with the LOS. In the other guidance technique, the vehicle speed is used as a means to formulate the guidance scheme. This is called a hybrid guidance system since it utilises various existing guidance laws during different phases of the mission as depicted in Figure 10.

Figure 10:

The idea is to gradually decrease the speed of the AUV as it approaches the cable/pipeline. Please refer to Naeem *et al.* (2004*b*) and Naeem *et al.* (2003) for a detailed description and evaluation of the pure pursuit and hybrid guidance strategies respectively. From an implementation perspective, it is currently not possible to test the hybrid guidance law in *Hammerhead* since the AUV model is only available at one fixed speed. However, the concept has been verified by borrowing an AUV dynamic model from the literature which is represented in terms of vehicle velocity. In general, the proposed schemes require the vehicle's speed and orientation to evaluate the guidance signals which are available from an IMU onboard the *Hammerhead*.

5 *HAMMERHEAD* AUTOPILOT DESIGN

Development of an autopilot for the *Hammerhead* AUV is of vital importance to the absolute design stage. This section presents controller design and results as applied to the *Hammerhead* AUV models identified in Section 3. It should be noted that there is a plethora of control systems available and a comprehensive review has been undertaken by Craven *et al.* (1998). However, the selection of a particular controller for an AUV is attributed to several factors. Some of them are

- Robustness to modelling errors (plant parameter variations)
- Disturbance handling characteristics
- Set point tracking and trajectory following
- Stability characteristics
- Application to linear and nonlinear plants

Two robust optimal control strategies and their variants have been selected as the candidate control schemes for the *Hammerhead* vehicle. A discrete-time LQG/LTR and the model based predictive controller which has been modified to accommodate various AI techniques for improved performance. Simulation and experimental results of the application of the proposed controllers to the *Hammerhead* vehicle in the horizontal and dive planes are presented.

5.1 LQG/LTR Controller Design

LQG is an optimal controller whose name is derived from the fact that it assumes a linear system, quadratic cost function and Gaussian noise. Unlike pole placement method, where the designer must know the exact pole locations, LQG places the poles at some arbitrary points within the unit circle so that the resulting system is optimal in some sense. A linear quadratic state feedback regulator (LQR) problem is solved which assumes that all states are available for feedback. However, this is not always true because either there is no available sensor to measure that state or the measurement is very noisy. A KF can be designed to estimate the unmeasured states. The LQR and KF can be designed independently and then combined to form an LQG controller, a fact known as the *separation principle*. Individually the LQR and KF have strong robustness properties with gain margin up to infinity and over $60(deg)$ phase margin, (Burl, 1999). Unfortunately, the LQG has relatively poor stability margins which can be circumvented by using LTR. The LTR works by adding fictitious noise to the process input which effectively cancels some of the plant zeros and possibly some of the stable poles, and inserts the estimator's zeros (Maciejowski, 1985; Skogestad and Postlethwaite, 1996). Herein, a discrete time LQG/LTR design has been developed motivated from the work of Maciejowski (1985). A substantial amount of material is available on the state feedback LQG/LTR controller, (see for example, Franklin *et al.* (1998) and Burl (1999)). Therefore, attention is focussed only on the application of the autopilot to the *Hammerhead* and no controller design details are presented.

5.1.1 LOS Following

The LQG/LTR controller has been tuned using the methodology proposed by Maciejowski (1985). In this technique, only the noise covariance matrices are required to be adjusted for the KF. The weighting matrices of the LQR are then selected according to an automatic procedure. See Naeem *et al.* (2003) for more details regarding the LQG/LTR autopilot design for *Hammerhead*. The algorithm is tested for a setpoint change in heading angle. The vehicle is assumed to be pointing in an arbitrary direction and is required to follow a certain heading angle closely without much control effort. A saturation block is inserted in series with the controller with cutoff limits of $\pm 20(deg)$ and a desired heading angle of $100(deg)$ is chosen with the vehicle initiating close to $0(deg)$. The measurement and process noise covariance matrices are adjusted to achieve the desired closed loop frequency response whereas weighting matrices for the objective function are selected according to Maciejowski (1985). The algorithm is simulated and the output response is depicted in Figure 11.

Figure 11:

The AUV heading bears a negligible overshoot and the settling time is less than $40(sec)$[1]. However, the price to pay for this settling time is that the actuator saturation constraint becomes active for about $35(sec)$ in the beginning of the simulation run when the vehicle was making a turn.

Increasing the control input weighting matrix can help limit the rudder movement within the constrained boundaries and avoid this saturation but at the cost of large settling times and deviation from the desired stability margins.

[1]since $Ts = 1$, therefore 1 sample time corresponds to 1 second

5.1.2 Depth Control

For depth controller design, the same procedure is adopted as before using the *Hammerhead* depth dynamic model. Simulations have been performed where the vehicle is assumed to be launched on the surface and the task is to follow a depth of $3(m)$ below the sea surface. The simulation is run for $100(sec)$ and the *Hammerhead* response is depicted in Figure 12. With a small overshoot of less than 5 per cent and settling time approximately $15(sec)$, the vehicle successfully follows the desired depth and stays on course throughout the rest of the mission duration.

Figure 12:

5.2 Model Predictive Control

MPC refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimize the future behaviour of a plant. Originally developed to meet the specialised control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace and metallurgy (Qin and Badgewell, 2000), to name but a few. A good account of MPC technology from the past to the future has been reviewed by Morari and Lee (1999), while a comparison between both theoretical and practical aspects of MPC has been undertaken by Carlos *et al.* (1989). For the interested reader, several other useful references on MPC can be found (Maciejowski, 2002; Clarke, 1994; Soeterboek, 1992; Richalet, 1993; Rawlings, 2000).

Herein, two modifications have been proposed to the standard predictive control problem. In the first technique, the conventional optimizer is replaced by a GA. One of the distinct advantages of using a GA is the possibility of employing various objective functions and the ability to deal with any type of process model and constraints, thus generalising a range of MPC technologies where each of them is defined on a fixed set of process model and objective function. In the other proposed strategy, a fuzzy performance index is used in place of the quadratic objective function and a GA is employed as an optimization tool. Herein, only GA-based MPC with a standard cost function will be discussed and some real-time results are elaborated. The reader is refered to Naeem *et al.* (2004*a*) for the fuzzy based GA-MPC autopilot design. The genetic-based control algorithm is depicted in Figure 13. As shown, the GA replaces the optimizer block and the AUV model identified from SI on the trials data has been used. The GA-based controller uses the process model to search for the control moves, which satisfy the process constraints and optimizes a cost function. A conventional quadratic objective function is minimised to evaluate the control inputs necessary to track a reference trajectory and is given by Equation (37).

Figure 13:

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \quad (37)$$

subject to input constraints

$$u^-_{constraint} \leq u(k) \leq u^+_{constraint}$$

$$\Delta u^-_{constraint} \leq \Delta u(k) \leq \Delta u^+_{constraint}$$

where the positive and negative signs represent the upper and lower constrained limits respectively. Q is the weighting scalar on the prediction error given by

$$e(k) = \hat{y}(k) - w(k) \quad (38)$$

where $w(k)$ is the reference or the desired setpoint. R and S are weights on the change in input, Δu and magnitude of input, u respectively. Please refer to Naeem *et al.* (2004*b*) for a comprehensive treatment on the GA-based MPC design.

The performance of the GA-MPC controller is next evaluated using the *Hammerhead* models identified in Section 3.1. Likewise LQG/LTR controller explicated in Section 5.1, separate controllers are designed for the yaw and depth channels. However, real-time results are shown here only for the horizontal plane whereas simulated data is illustrated for the depth output. It is important to point out the fact that the authors tested the controller on a 1960s made vehicle hull with the bulk of the existing electronics specifically the motors and their mechanical assembly being retained. Due to this, the rudder movement could not be controlled precisely and the minimum deflection observed was $2(deg)$. For this reason, a minimum rate of change of input constraint was imposed on the rudder which could lead to chattering effects in the rudder movement.

5.2.1 Heading Control

To simulate the controller, it is assumed that there is no model/plant mismatch. The constraints on the rudder were $\pm 20(deg)$ while the minimum allowed deflection is $2(deg)$ as discussed in the previous section. The weighting matrices were adjusted heuristically and the prediction and control horizons (H_p and H_c) of MPC were set to minimise the control effort and to increase the speed of response.

The controller was simulated for a step change in heading and the result is depicted in Figure 14. Without taking any disturbance into account, the vehicle closely follows the desired course after initiating from an arbitrary direction. The response in Figure 14(b) bears a small overshoot which can be minimised by adjusting the weighting matrices but at the cost of slower response time. The rudder deflections generated by the GA-based controller are also shown in Figure 14(a) requiring minimum control effort and stays within the specified bounds. There is a large movement in the rudder position around $t = 50(sec)$ yet this does not affect the vehicle's motion because of its slow dynamics. The spike is due to the probabilistic nature of GA which produces such results unless accounted for in the code but has not been implemented owing to the extra computational burden that it imposes. The chattering phenomena can also be observed from this figure because of the $2(deg)$ rate constraint. However, the effect of this is almost negligible on the vehicle's movement.

Figure 14:

Next, the heading controller was tested in the *Hammerhead*. The parameters used in these trials were kept the same as in the simulation studies for a fair comparison. The test was carried out for a step change in heading where the initial and desired headings were the same as in the simulation. It is evident from Figure 15(b) that the GA-MPC was able to track the desired heading without any offset despite the presence of always existing model uncertainty and external

disturbances. There is an overshoot though which could be blamed on the surface currents. Looking at the rudder deflections in Figure 15(a), again there is some expected chattering present. However, the rudder movement always remain within the specified constraints. There is a large spike followed by fluctuations in rudder movement at approximately $125(sec)$ in response to the change in vehicle's heading due to surface currents. However, the controller is robust enough to cope with it and attains the steady state input and output values in approximately $50(sec)$. A statistical analysis reveals that the standard deviation of rudder deflections in experimental data is approximately $11(deg)$, while it is quite high $(8(deg))$ in the simulation (even without any disturbance), possibly because of the chattering phenomena.

Figure 15:

It is interesting to note that the rise time in the experimental data is much smaller $(24(sec))$ than in the simulation $(38(sec))$. One reason for this is the effect of surface currents pushing the vehicle unwantingly and causing even a higher overshoot as compared to the simulated response. Model/plant mismatch could also be a potential source of this problem.

5.2.2 Depth Control

A GA-based MPC depth autopilot is now developed for *Hammerhead* in this section. The model used for controller design is given by Equation (36) with a sampling time of $1(sec)$ and identified from trials data using SI techniques. The intent is to design a controller that is able to control the depth of the AUV as closely as possible despite the presence of external disturbances and modelling errors. The performance of the control strategy developed herein will eventually be assessed on the *Hammerhead* vehicle.

The front canards movement is restricted to $\pm 25(deg)$ which was thought to be adequate to control the vehicle in the vertical plane and was obtained through a series of rigorous in water experiments. Please note that no minimum rate constraint is imposed on the hydroplane movement since these are newly installed in the vehicle. It is assumed that the vehicle is manouevring near the surface (zero depth) and is subjected to a depth command of $3(m)$. The *Hammerhead* response to a step change in depth when diving from the sea surface is shown in Figure 16.

Figure 16:

It took less than $25(sec)$ for the vehicle to attain the desired depth of $3(m)$ with little overshoot and no steady state error. The diving rate stays uniform as the vehicle approaches the desired level with precision and maintains the specified depth throughout.

6 CONCLUDING REMARKS

This chapter presents an overview of the navigation, guidance and control system design of *Hammerhead* vehicle. Two navigation scenarios have been considered to validate the proposed approach: 2D/surface and 3D/surface-depth scenarios. In both scenarios, the data from the TCM2 electronic compass and IMU are fused with two other simulated sensors before being used in transforming data from the body to the NED co-ordinate frame, where integration between the INS and GPS data occurs. In the first scenario, as the vehicle operates on the surface only, the GPS data is available periodically and the proposed estimation process takes place between the GPS fixes. In the second scenario, the GPS fixes are available continuously when the vehicle

operates on the surface, and the proposed estimation algorithm blends these data with the position solution produced by the dead reckoning method to find the best estimates of the vehicle's position. In this scenario, the vehicle uses only dead reckoning method during an underwater mission and the accumulated errors produced thereby is reset by GPS fixes the next time the vehicle gain access to their signals. It has been shown in both scenarios that the proposed algorithm has produced a significant improvement in accuracy and reliability of the navigation system of the vehicle.

The use of SI techniques on actual vehicle data for modelling is suggested where the models are subsequently used for autopilot development. Since *Hammerhead* is a cruising type vehicle, therefore guidance laws specifically for cable/pipeline inspection task are formulated. Finally, autopilots based on optimal control and AI techniques are conceived. Some real-time results based on GA-MPC controller are presented and a comparison is made with simulation examples. It is shown that the proposed autopilot is quite robust under a varying set of conditions including external disturbances and modelling uncertainty.

It is envisaged that the NGC techniques employed for *Hammerhead* which are presented within this chapter will be considered as an invaluable addition to the underwater research literature.

References

Ahmad, S. M. and R. Sutton (2003). Dynamic Modelling of a Remotely Operated Vehicle. In: *Proceedings 1st IFAC Workshop on Guidance and Control of Underwater Vehicles GCUV 2003*. IFAC. Newport, South Wales, UK. pp. 47–52.

Asakawa, K., J. Kojima, Y. Kato, S. Matsumoto and N. Kato (2000). Autonomous Underwater Vehicle *Aqua Explorer 2* for Inspection of Underwater Cables. In: *Proceedings of the 2000 International Symposium on Underwater Technology*. Tokyo, Japan. pp. 242–247.

Burl, J. B. (1999). *Linear Optimal Control, H_2 and H_∞ Methods*. Addison-Wesley Longman Inc.

Carlos, E. G., D. M. Prett and M. Morari (1989). Model Predictive Control: Theory and Practice - a Survey. *Automatica* **25**(3), 335–348.

Clarke, D., Ed.) (1994). *Advances in Model-Based Predictive Control*. Oxford Science Publications.

Craven, P. J., R. Sutton and R. S. Burns (1998). Control Strategies for Unmanned Underwater Vehicles. *The Journal of Navigation* **51**(1), 79–105.

Dalgleish, F. R. (2004). Applications of Laser-Assisted Vision to Autonomous Underwater Vehicle Navigation. PhD Thesis. Cranfield University. Cranfield, UK.

Dalgleish, F. R., S. Tetlow and R. L. Allwood (2003). A Preliminary Experiments in the Development of a Laser Based-Imaging Sensor for AUV Navigation. In: *Proceedings 1st IFAC Workshop on Guidance and Control of Underwater Vehicles*. Newport, South Wales, UK. pp. 239–244.

Escamilla-Ambrosio, P. J. and N. Mort (2001). A Hybrid Kalman Filter-Fuzzy Logic Multisensor Data Fusion Architecture with Fault Tolerant Characteristics. In: *Proceedings of the 2001 International Conference on Artificial Intelligence*. Las Vegas, NV, USA. pp. 361–367.

Fitzgerald, R. J. (1971). Divergence of the Kalman Filter. *IEEE Transactions on Automatic Control* **AC-16**(6), 736–747.

Franklin, G. F., J. D. Powell and M. Workman (1998). *Digital Control of Dynamic Systems*. 3rd ed.. Addison-Wesley Longman Inc.

Gade, K. and B. Jalving (1999). An Aided Navigation Post Processing Filter for Detailed Seabed Mapping UUVs. *Modeling, Identification and Control* **20**(3), 165–176.

Goheen, K. R. and E. R. Jefferys (1990). The Application of Alternative Modelling Techniques to ROV Dynamics. In: *Proceedings IEEE International Conference on Robotics and Automation*. IEEE. Cincinnati, OH, USA. pp. 1302–1309.

Grenon, G., P.E. An, S.M. Smith and A.J. Healey (2001). Enhancement of the Inertial Navigation System for the Morpheous Autonomous Underwater Vehicles. *IEEE Journal of Oceanic Engineering* **26**(4), 548 – 560.

Ippoliti, C. G., S. Radicioni and A. Rossolini (2001). Multiple Models Control of a Remotely Operated Vehicle: Analysis of Models Structure and Complexity. In: *Proceedings IFAC Conference on Control Applications in Marine Systems (CAMS' 01)*. IFAC. Glasgow, Scotland, UK.

Jetto, L., S. Longhi and D. Vitali (1999). Localisation of a Wheeled Mobile Robot by Sensor Data Fusion Based on a Fuzzy Logic Adapted Kalman Filter. *Control Engineering Practice* **7**, 763–771.

Kinsey, J.C. and L.L. Whitcomb (2003). Preliminary Field Experience with the DVLNAV Integrated Navigation System for Manned and Unmanned Submersibles. In: *Proceedings of the 1st IFAC Workshop on Guidance and Control of Underwater Vehicles*. Newport, South Wales, UK. pp. 83–88.

Kobayashi, K., K. C. Cheok, K. Watanabe and F. Munekata (1998). Accurate Differential Global Positioning via Fuzzy Logic Kalman Filter Sensor Fusion Technique. *IEEE Transaction on Industrial Electronics* **45**(3), 510–518.

Ljung, L. (1999). *System Identification, Theory for the User*. 2^{nd} ed.. PTR Prentice Hall.

Loebis, D. (2004). An Intelligent Navigation System for an Autonomous Underwater Vehicle. PhD Thesis. The University of Plymouth. Plymouth, UK.

Loebis, D., F. R. Dalgleish, R. Sutton, S. Tetlow, J. Chudley and R. Allwood (2003). An Integrated Approach in the Design of Navigation System for an AUV. In: *Proceedings of MCMC 2003 Conference*. Girona, Spain. pp. 329–334.

Loebis, D., R. Sutton and J. Chudley (2002). Review of Multisensor Data Fusion Techniques and Their Application to Autonomous Underwater Vehicle Navigation. *Journal of Marine Engineering and Technology* **AC-15**(2), 175–184.

Maciejowski, J. M. (1985). Asymptotic Recovery for Discrete-Time Systems. *IEEE Transactions on Automatic Control* **AC-30**(6), 602–605.

Maciejowski, J. M. (2002). *Predictive Control with Constraints*. Prentice hall.

Mehra, R. K. (1970). On the Identification of Variances and Adaptive Kalman Filtering. *IEEE Transactions on Automatic Control* **AC-15**(2), 175–184.

Mohamed, A. H. and K. P. Schwarz (1999). Adaptive Kalman Filtering for INS/GPS. *Journal of Geodesy* **73**, 193–203.

Morari, M. and J. M. Lee (1999). Model Predictive Control: Past, Present and Future. *Computers and Chemical Engineering* **23**, 667–682.

Naeem, W. (2004). Guidance and Control of an Autonomous Underwater Vehicle. PhD Thesis. The University of Plymouth. Plymouth, UK.

Naeem, W., R. Sutton and J. Chudley (2003). LQG/LTR Control of an Autonomous Underwater Vehicle Using a Hybrid Guidance Law. In: *Proceedings of Guidance and Control of Underwater Vehicles 2003*. Elsevier IFAC Publications. Newport, South Wales, UK. pp. 31–36.

Naeem, W., R. Sutton and J. Chudley (2004*a*). Model Predictive Control of an Autonomous Underwater Vehicle with a Fuzzy Objective Function Optimized Using a GA. In: *Proceedings of Control Applications in Marine Systems (CAMS'04)*. IFAC. Ancona, Italy. pp. 433–438.

Naeem, W., R. Sutton and S. M. Ahmad (2004*b*). Pure Pursuit Guidance and Model Predictive Control of an Autonomous Underwater Vehicle for Cable/Pipeline Tracking. *IMarEST Journal of Marine Science and Environment* **C**(1), 25–35.

Naylies, I. (2000). The Sensory Requirement of a PC Controlled AUV. Master's thesis. Offshore Technology Centre, Cranfield University.

Qin, S. Joe and Thomas A. Badgewell (2000). An Overview of Nonlinear Model Predictive Control Applications. In: *Nonlinear Model Predictive Control*. Switzerland.

Rawlings, J. B. (2000). Tutorial Overview of Model Predictive Control. *IEEE Control Systems Magazine* pp. 38–52.

Richalet, J. (1993). Industrial Applications of Model Based Predictive Control. *Automatica* **29**(5), 1251–1274.

Skogestad, S. and I. Postlethwaite (1996). *Multivariable Feedback Control: Analysis and Design Using Frequency-Domain Methods*. John Wiley and Sons Ltd.

Soeterboek, R. (1992). *Predictive Control, A Unified Approach*. Prentice Hall.

Størkersen, N., J. Kristensen, A.Indreeide, J. Seim and T. Glancy (1998). Hugin - UUV for Seabed Surveying. *Sea Technology* **39**(2), 99–104.

Tetlow, S. and R. L. Allwood (1995). Development and Applications of a Novel Underwater Laser Illumination System. *Underwater Technology* **21**(2), 13–20.

Tinker, S. J., A. R. Bowman and T. B. Booth (1979). Identifying Submersible Dynamics from Free Model Experiments. In: *RINA Annual Report and Transactions*. pp. 191–196.

Wright, J., K. Scott, C. Tien-Hsin, B. Lau, J. Lathrop and J. McCormick (1996). Multi-Sensor Data Fusion for Seafloor Mapping and Ordnance Location. In: *Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology*. Monterey, CA, USA. pp. 167–175.

Yun, X., R.E. Bachmann, R.B McGhee, R.H. Whalen, R.L. Roberts, R.G. Knapp, A.J. Healey and M.J. Zyda (1999). Testing and Evaluation of an Integrated GPS/INS System for Small AUV Navigation. *IEEE Journal of Oceanic Engineering* **24**(3), 396 – 404.

$\mathbf{R}_k$ $\lvert\mathbf{delta}\rvert_k$	Z	S	L
Z	G	G	AV
S	G	AV	P
L	AV	P	P

Table 1: Fuzzy rule based FLO

Limit	Parameter		
	D_3, M_1	D_4, M_2, I_1	M_3, I_2
Upper Limit	-0.135	-0.033	0.033
Lower Limit	-0.033	0.033	0.135

Table 2: FKF boundaries

Parameters	Values
Number of objective functions	5
Number of generation	25
Number of individual per generation	10
Generation gap in selection operation	0.95
Rate in rate in recombination operation	0.8
Rate in mutation operation	0.09

Table 3: MOGA parameters

Sensor	$J_{zv}(rad)$	$J_{ze}(rad)$	
		Non-MOGA	MOGA
Sensor-1	0.2340	0.2090	0.2094
Sensor-2	0.2960	0.3047	0.2761
Sensor-3	0.6558	0.4131	0.4130
Sensor-4	0.3852	0.2552	0.2551
Fused		0.2487	0.2088

Table 4: Comparison of performance

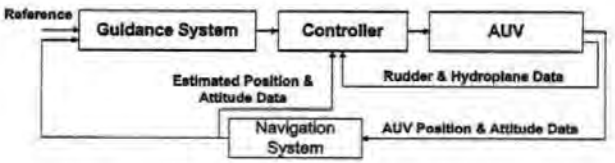
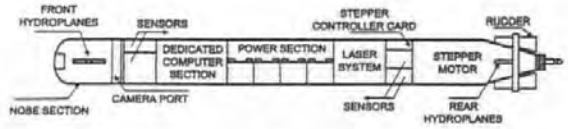


Figure 1: Navigation, guidance and control for the *Hammerhead* AUV



(a)

(b)

Figure 2: (a) The schematic of the *Hammerhead*, (b) The *Hammerhead* strapped on its trailer

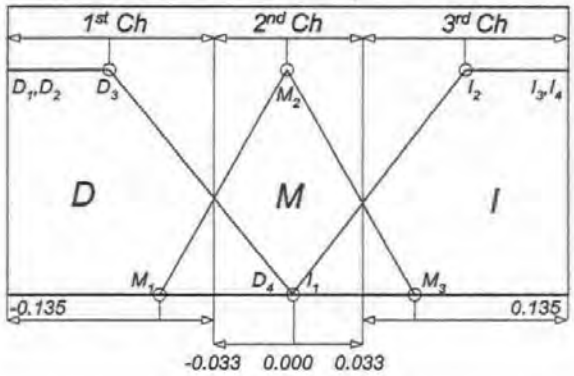
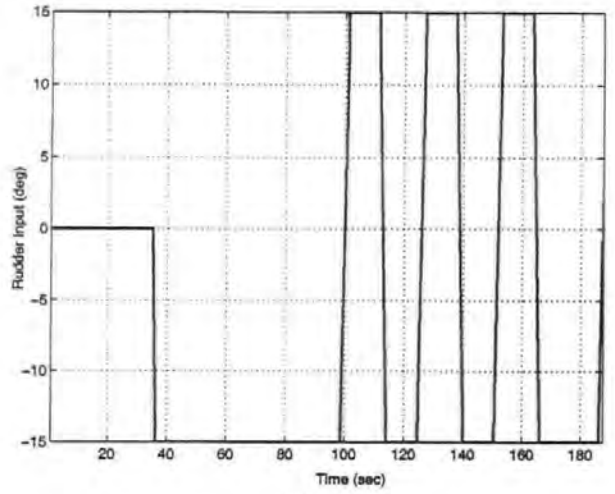
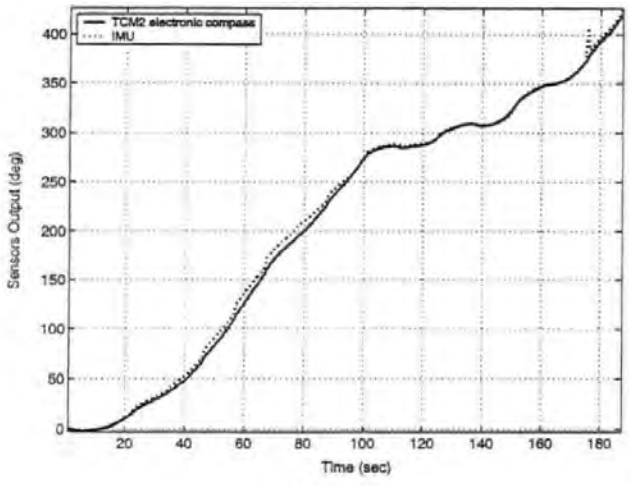


Figure 3: Membership function and boundaries of $\mathbf{R}_k$

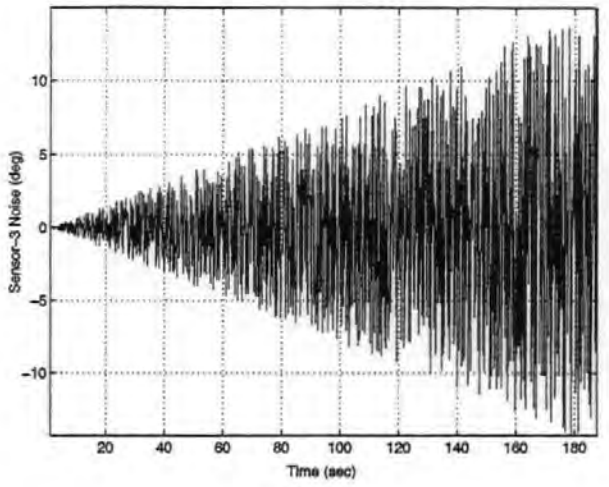


(a)

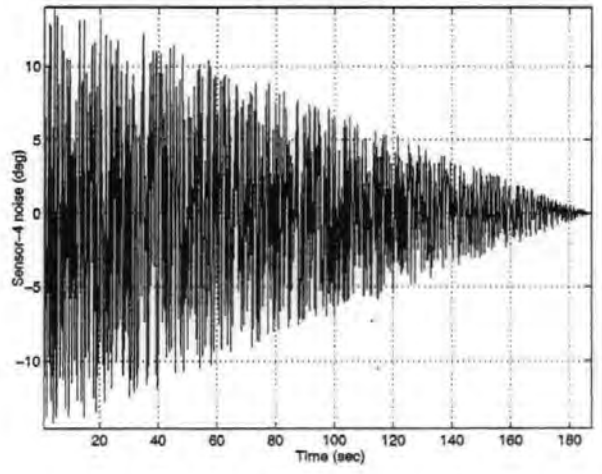


(b)

Figure 4: (a) Rudder input, (b) TCM2 electronic compass and IMU output



(a)



(b)

Figure 5: (a) Sensor-3 noise, (b) sensor-4 noise

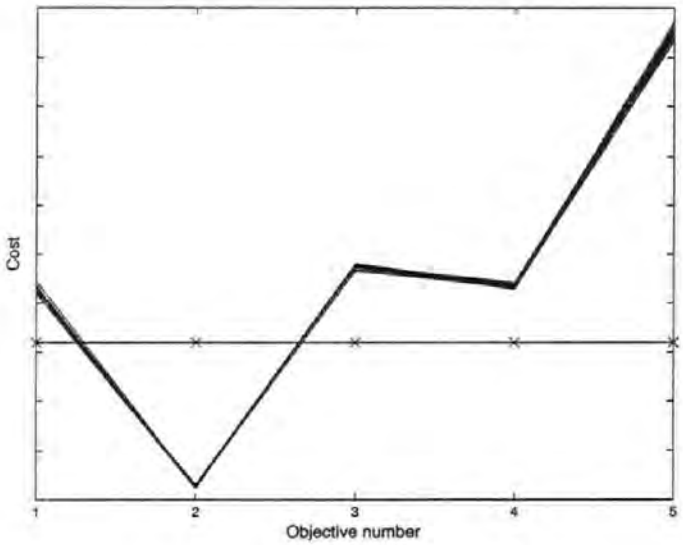
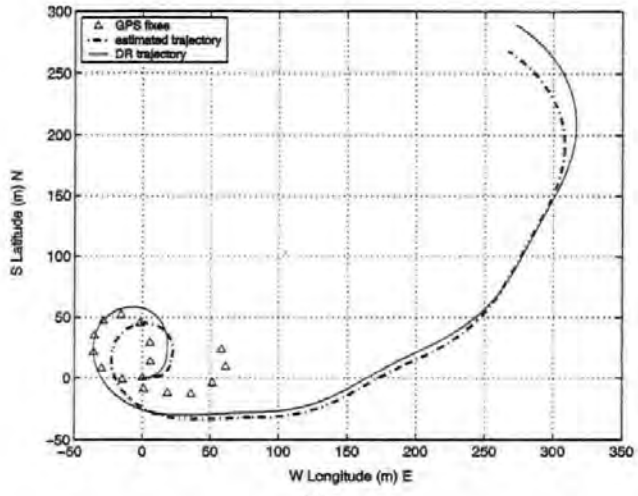
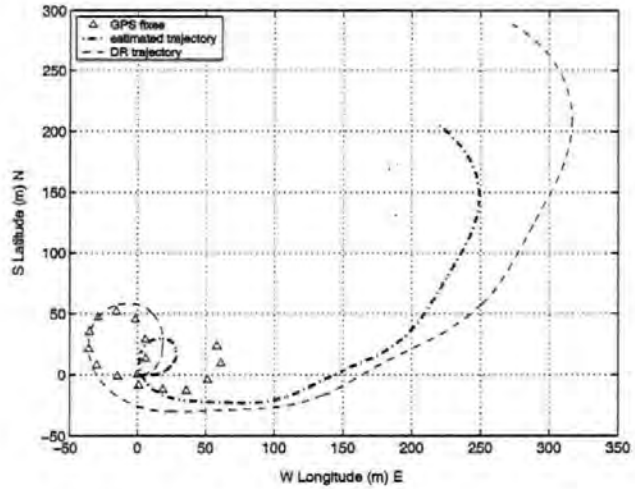


Figure 6: Trade-off graphs for the FKF search



(a)



(b)

Figure 7: (a) AUV trajectory obtained using GPS, INS sensors (dead reckoning method) and GPS/INS using EKF without adaptation, (b) AUV trajectory obtained using GPS, INS sensors (dead reckoning method) and GPS/INS using EKF with adaptation

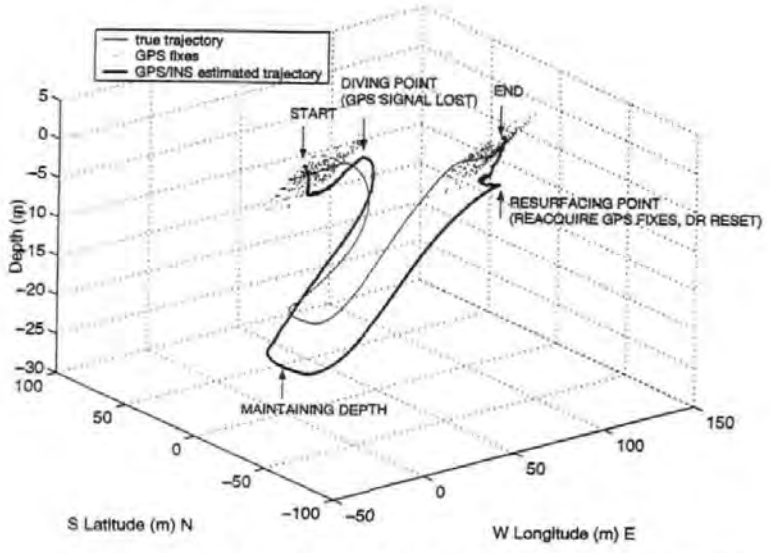


Figure 8: True trajectory, GPS fixes and GPS/INS using yaw produced by sensor-4 only

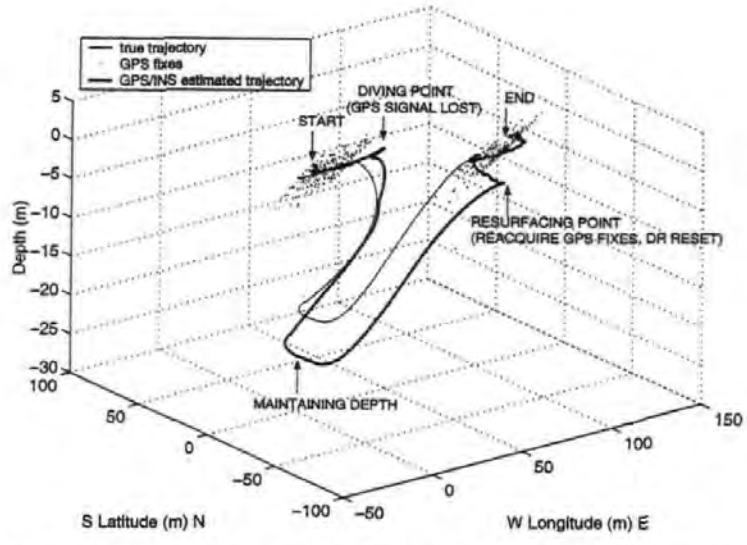


Figure 9: True trajectory, GPS fixes and GPS/INS using yaw produced by fused sensor

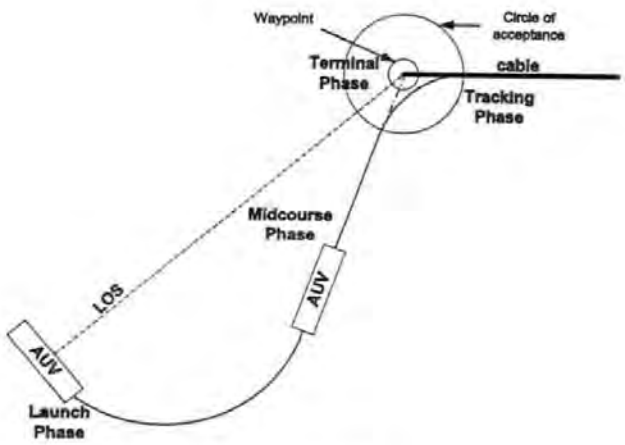


Figure 10: Planar view of the four phases of flight for cable tracking problem of an AUV.

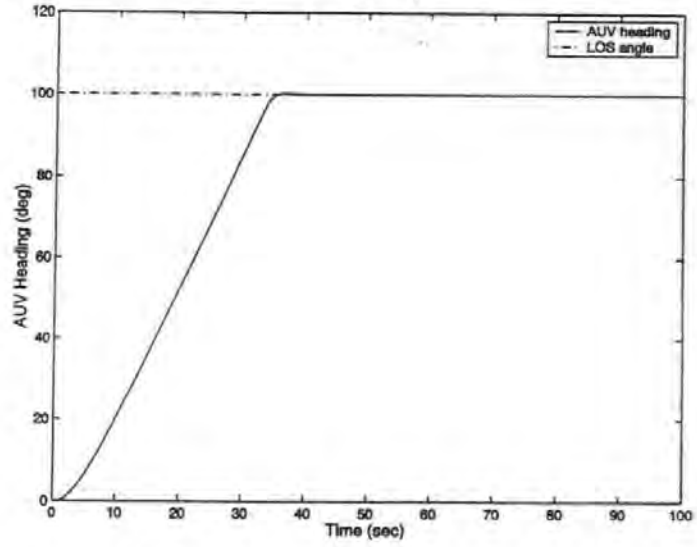


Figure 11: LQG/LTR control of *Hammerhead* showing LOS tracking.

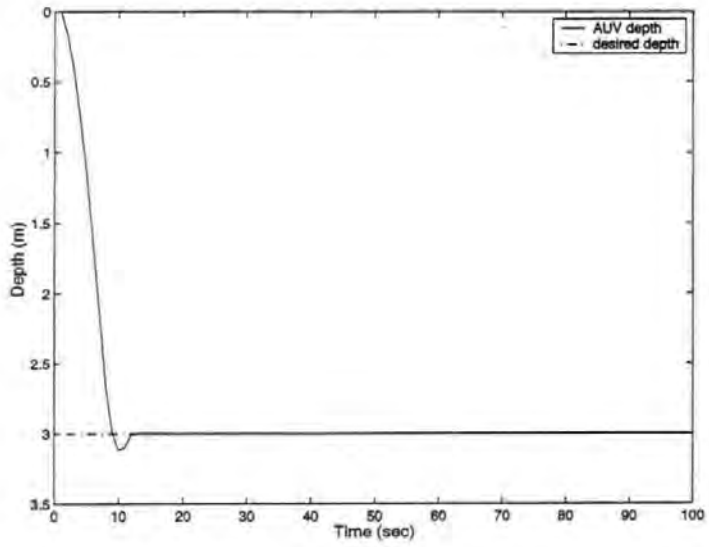


Figure 12: Depth control of *Hammerhead* vehicle using the LQG/LTR controller showing a step change in depth.

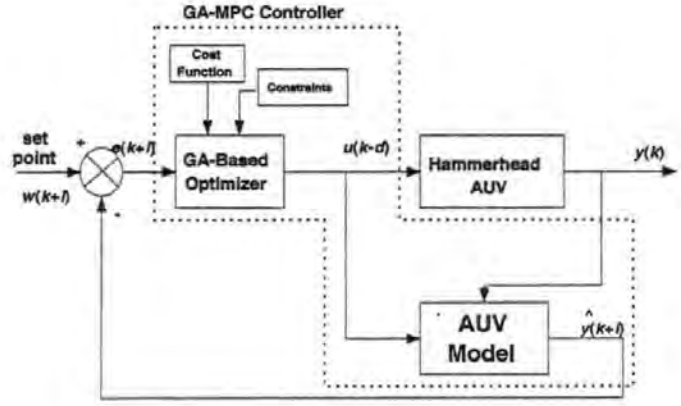


Figure 13: Genetic algorithm based model predictive controller.

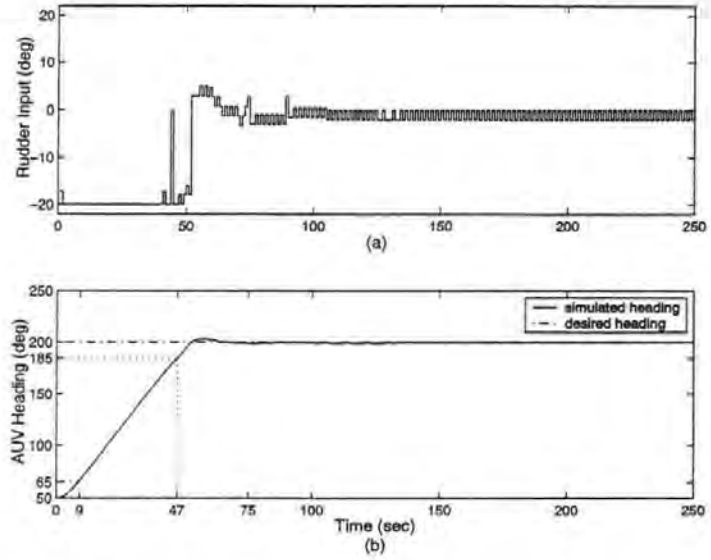


Figure 14: GA-MPC simulation results (a) Rudder deflections generated by the controller (b) AUV heading.

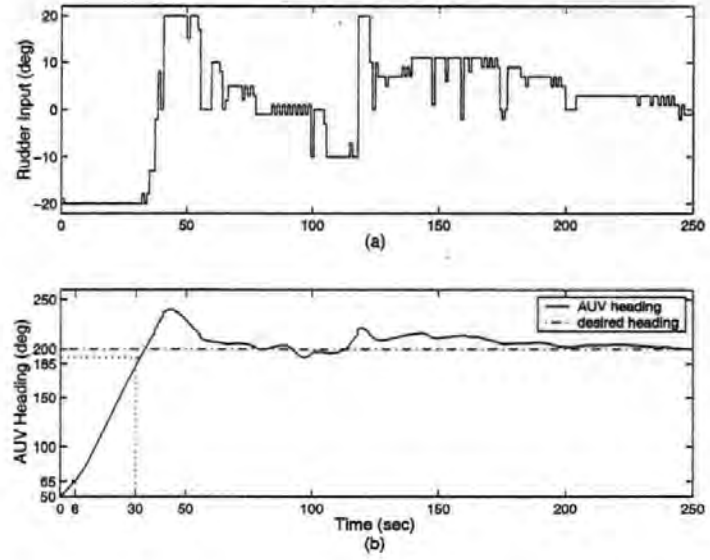


Figure 15: Controller trial results (a) Rudder deflections generated by the GA−MPC (b) *Hammerhead* heading obtained from IMU.

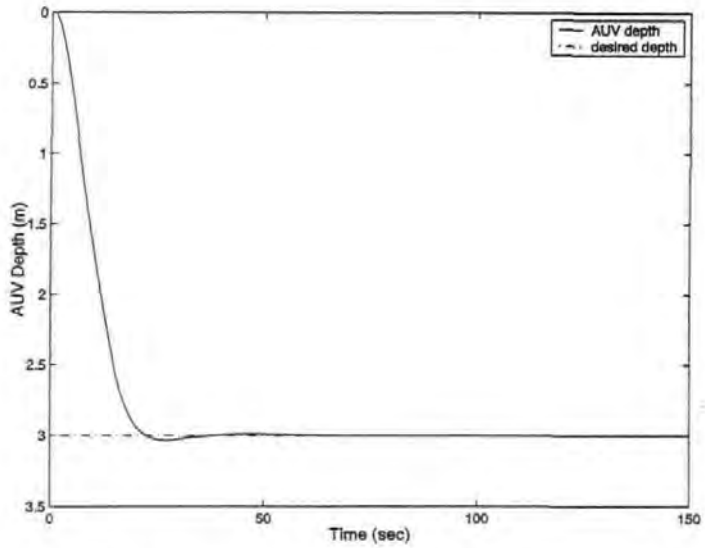


Figure 16: *Hammerhead* step response for a change in depth obtained by employing a GA-MPC depth autopilot

A genetic algorithm-based model predictive control autopilot design and its implementation in an autonomous underwater vehicle

W Naeem[1]*, R Sutton[1], J Chudley[1], F R Dalgleish[2] and S Tetlow[2]
[1]Marine and Industrial Dynamic Analysis Research Group, University of Plymouth, Plymouth, UK
[2]Offshore Technology Centre, Cranfield University, Cranfield, UK

Abstract: The control of any underwater vehicle has always been a challenging task and is certainly an important and necessary feature of an autonomous underwater vehicle (AUV). This paper describes the implementation of a genetic algorithm (GA)-based model predictive controller for an AUV named *Hammerhead* which is being developed jointly by the Universities of Plymouth and Cranfield. To the present authors' knowledge, this is the first successful application of a GA in real-time optimization for controller tuning in the marine sector and thus the paper makes an extremely novel and useful contribution to control system design. The advantages of using model predictive control (MPC) include its constraint handling and disturbance rejection properties commonly present in an underwater environment. The use of GAs generalizes MPC to employ linear as well as non-linear process models. Furthermore, it supports the inclusion of various types of objective function without having to change the controller structure. The model required for MPC is extracted using system identification techniques on actual AUV data obtained from full-scale in-water tests. A description of *Hammerhead* AUV is outlined and simulation and experiment data are shown which were obtained by optimizing the cost function online using the GA. Results demonstrate good tracking behaviour despite the presence of disturbances and ever-present modelling uncertainty.

Keywords: autonomous underwater vehicles, system identification, model predictive control, genetic algorithm

NOTATION

d	disturbance
e	prediction error
H_c	control horizon
H_p	prediction horizon
J	performance index or criterion function
Q	weight on the prediction error
R	weight on the change in input
S	weight on the input
u	input to the system
u_{opt}	optimal controller output sequence
w	reference trajectory or set point
y	output or response of the system
$\hat{y}$	predicted process output from the model
Δu	change in the input u
ε	residuals

The MS was received on 30 September 2003 and was accepted after revision for publication on 26 May 2004.
** Corresponding author: Marine and Industrial Dynamic Analysis Research Group, University of Plymouth, Drake Circus, Plymouth, PL4 8AA, UK. email: w.naeem@plymouth.ac.uk*

1 INTRODUCTION

Underwater robots are on the verge of technological advances. There have been a myriad of efforts for underwater vehicles system design in terms of their physical structure and the controller/sensors used on board. Most of the vehicles are designed with a specific application in mind. The applications can range from subsea cable/pipeline tracking to dam inspection and mine clearing operations. Generally, there are two types of unmanned underwater vehicle, namely remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs). ROVs require a trained human operator for manoeuvring and are generally attached to an umbilical as a source of power and for communication. In this regard, ROVs have limited capabilities mainly because of the umbilical

length. On the other hand, AUVs do not have a tether and the range of a mission is only limited by the on-board batteries. AUVs are viable tools for underwater exploration in places where human divers cannot reach or human life is put in jeopardy such as surveying an underwater volcano. To be able to perform accurately its required task, the AUV needs to have a reliable navigation, guidance and control system. The design of an autopilot (control system) for an underwater vehicle generally requires a mathematical model of the physical system. This is imperative in gaining physical insight concerning the behaviour of the system.

The Universities of Plymouth and Cranfield together are developing a low-cost AUV named *Hammerhead* to provide an easy-to-use platform for testing various control algorithms and for use by other underwater research groups in the UK. Some of the numerous applications that are kept in mind for this particular AUV are cable/pipeline and sea bed inspection and object tracking. In addition, the present authors sense a distinct lack of information in the public domain on in-water trials and that there is a need for experimental tests for AUV autopilots. Lea [1] reviewed various control systems for AUVs, presenting experimental comparative tests, while Craven *et al.* [2] reported their findings on classical and advance controllers for AUVs.

It has been observed that, for practical applications, the majority of AUVs around the world employ simple control algorithms such as the proportional–integral–derivative (PID) controller. Clearly, this strategy is useful in commercializing the product in a short time by employing a controller that is not so easy to tune and that can be realized in hardware in a straightforward manner. For instance, the Natural Environmental Research Council's AUTOSUB vehicle based at the Southampton Oceanography Centre, UK, uses proportional–derivative (PD) controllers to control the vehicle position, depth and altitude which is shown to be adequate [3]. The underlying philosophy for selecting the PD controller is 'to keep it simple' for the delivery of a reliable and maintainable system. However, it is shown that the controller could not cope with changes in water density (resulting from a combination of fresh and saline water), which results in the loss of vehicle buoyancy, causing it to sink [4]. Similarly, the AQUA explorer [5] developed by KDD Research and Development Laboratories, Japan, and the Twin Burger [6] vehicle based at the University of Tokyo, employed PID and PD control algorithms for attitude control. This does not prevent researchers from developing a new breed of control laws suitable for underwater environment which are robust and overcome many difficulties inherent in PID controllers. For example, in addition to PID control, a fuzzy control law is developed for the AQUA explorer and experimental results for the case of a cable tracking mission have been reported [7, 8]. A neural network-based control system has also been implemented successfully in the Twin Burger vehicle [9]; however, the online implementation entails high computing power or parallel processing to obtain the controller parameters in real time. Aguiar and co-workers [10, 11] have realized a sliding mode controller in the SIRENE underwater vehicle based at Instituto Superior Técnico, Portugal. The controller provides good robustness to modelling errors and has essentially a PID-type structure with additional non-linear terms to account for vehicle parametric uncertainty. Another sliding mode control methodology was developed by Healey and Lienard [12] at the Naval Post Graduate School (NPS), Monterey, California. The NPS ARIES AUV is a successful example of the implementation of the sliding mode technique to control the vehicle in the horizontal and vertical planes [13].

This paper is concerned with system identification (SI) and control of an AUV. The SI approach is useful in providing reliable and accurate models in a short time without relying too much on mathematical modelling techniques. This feature therefore, is attractive for the underwater vehicle manufacturers, where a vehicle configuration changes frequently to suit the mission requirements. AUV modelling using SI approaches have been investigated before [14–18], but most of the work involved has been done on identifying a model by generating data from a six-degrees-of-freedom mathematical model of the vehicle. Moreover, mathematical modelling requires the hydrodynamic coefficients of the AUV to be estimated (see, for example, reference [19]), which is very demanding and expensive and requires a special platform that is not available everywhere. However, in this paper, the SI is performed on actual AUV input/output data obtained from test trials explained in the next section.

The control system developed is a genetic algorithm (GA)-based model predictive control (MPC). MPC was originated and has long been used in process industry because of its strong robustness and constraint handling characteristics. It is the only methodology that can handle constraints in a natural way during controller design. While the traditional controllers cope with constraints in an *ad hoc* way during the implementation phase, the model predictive controller is fully aware of its boundaries and therefore is able to operate close to the constraints without actually violating them. As far as the present authors are concerned, MPC on AUVs was first simulated by Kwiesielewicz *et al.* [20] who compared its performance to PD control and an adaptive neurofuzzy inference system-tuned autopilot. The results were found to be quite promising. Then, Naeem and co-workers simulated GA-tuned MPC on an AUV for heading control [21] and for subsea cable/pipeline tracking [22]. The results demonstrated the robustness of GA-based MPC in the presence of sea currents.

Furthering the previous work on GA-based MPC as an autopilot, herein the present authors have implemented the said controller in real time in *Hammerhead* and thus this represents a new application area for this type of

control algorithm. In general, it is usually thought that a GA is computationally demanding and therefore not suitable for real-time processes. In the literature, there have been only a few applications, in general, demonstrating the use of an online GA (see, for example, references [23] and [24]). However, it has been successfully shown here that keeping the computational burden at its minimum, GA can be used as an efficient optimization technique for real-time control problems. The model employed by MPC has been identified first from test trials, which is then used as an integral part of the GA-based MPC. The following section gives a description of *Hammerhead* and is then followed by an exposé to SI in general in section 3. Section 4 details basic MPC and problem formulation using GA-based MPC is outlined in section 5. The experiment design for SI and the controller is explained in section 6. Problems encountered before and during the trials are also covered. Simulation results and experimental data are presented in section 7 showing results from SI trials and controller testing for a step change in heading. Finally, the paper ends with concluding remarks in section 8.

2 *HAMMERHEAD* SYSTEM DESCRIPTION

The *Hammerhead* vehicle used in this study was acquired from the Royal Navy some years ago, where it had previously been used as a deep mobile target. It has a torpedo-shaped body about $3\frac{1}{2}$ m long and approximately $\frac{1}{3}$ m in diameter. The control surfaces are the two rear rudders for steering and two front hydroplanes for diving. The rudder and hydroplanes are controlled by two separate on-board stepper motors and the command to the stepper motors is currently sent via an umbilical attached to the rear end of the vehicle. The other end of the umbilical is connected to a control computer running MATLAB on a surface vessel. The on-board sensors include an inertial measurement unit (IMU) containing three accelerometers and three rate gyros, TCM2 electronic compass, pressure sensor, global positioning system (GPS) and a shaft speed encoder. The data logged using the above-mentioned sensors is summarized in Table 1.

A laser stripe illumination system [25, 26], being developed at Cranfield University will also be implemented in the vehicle to enhance the navigation multi-sensor data fusion (MSDF) algorithm [27], which is being developed at the University of Plymouth. However, for the results to be presented in this paper, the data from the sensors mentioned in Table 1 were obtained and the IMU provides feedback for the heading controller. A sectional view of *Hammerhead* is shown in Fig. 1.

3 SYSTEM IDENTIFICATION

SI of a dynamic system generally consists of the following four steps:

Step 1. Data acquisition.

Step 2. Characterization.

Step 3. Identification/estimation.

Step 4. Verification.

The first and most important step is to acquire the input/output data of the system to be identified. Acquiring data is not trivial and can be very laborious and expensive. This involves careful planning of the inputs to be applied so that sufficient information about the system dynamics is obtained. If the inputs are not well designed, then it could lead to insufficient or even useless data.

Step 2 defines the structure of the system, e.g. type and order of the differential equation relating the input to the output. This means the selection of a suitable model structure, e.g. autoregressive with exogenous input (ARX), autoregressive moving average with exogenous input (ARMAX), output error (OE), etc.

Table 1 Logged sensors data

Sensors	Data
IMU	Heading, pitch, roll, linear and angular velocities
TCM2 compass	Heading, pitch and roll
Pressure sensor	Depth of the vehicle
GPS	Coordinates of the vehicle on the surface, forward speed
Shaft speed encoder	Vehicle speed

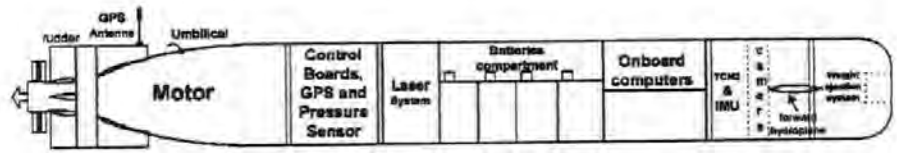


Fig. 1 Sectional view of *Hammerhead* showing sensor locations

Step 3 is identification/estimation, which involves determining the numerical values of the structural parameters, which minimize the error between the system to be identified, and its model. Common estimation methods are the least-squares method, the instrumental-variable method, the maximum-likelihood method and the prediction-error method.

The final step 4, verification, consists in relating the system to the identified model responses in the time or frequency domain to instil confidence in the obtained model. Residual (correlation) analysis and cross-validation tests have been employed for model validation in this paper. The above-mentioned features of SI are symbolically indicated in Fig. 2.

In this figure, u is the input, y is the output or response, d is the disturbance, $\hat{y}$ is the response of the model to the same input u and ε is the error between the model output and plant output also called the residuals. The objective of identification is to minimize the sum-squared errors or residuals ε. More details on SI can be found in reference [28].

4 MODEL PREDICTIVE CONTROL

MPC refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimize the future behaviour of a plant. Originally developed to meet the specialized control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace and metallurgy [29], to name but a few.

The development of MPC can be traced back to 1978 after the publication of the paper by Richalet *et al.* [30]. They named their algorithm model predictive heuristic control and it was successfully applied to a fluid catalytic cracking unit main fractionator column, a power plant steam generator and a poly(vinyl chloride) plant. Then, Cutler and Ramaker from Shell Oil Company in 1979 and 1980 developed their own independent MPC technology referred to as dynamic matrix control [31], and they showed results from a furnace temperature control application to demonstrate improved control quality. However, another form of MPC, called generalized predictive control has been devised by Clarke *et al.* [32, 33], and is employed in this paper. The fundamental difference between all these techniques is the type of model used and the cost function being optimized.

As noted above, the success that MPC is enjoying is attributed to the fact that it was developed in the industry, by the industry and for the industry. A good account of MPC technology from the past to the future has been reviewed by Morari and Lee [34], while a comparison between both theoretical and practical aspects of MPC has been undertaken by Carlos *et al.* [35]. For the interested reader, several other useful references on MPC can be found [36–40].

The process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the output of the process can be used. Further, if the process is subject to disturbances, a disturbance or noise model can be added to the process model. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used. Typically the criterion is the difference between the predicted process output and the desired reference trajectory. A simple criterion function is

$$J = \sum_{i=1}^{N_p} [\hat{y}(k+i) - w(k+i)]^2 \qquad (1)$$

where $\hat{y}$ is the predicted process output, w is the reference trajectory and H_p is the prediction horizon or output horizon. The structure of a model predictive controller is shown in Fig. 3.

The optimal controller output sequence u_{opt} over the prediction horizon is obtained by minimization of J with

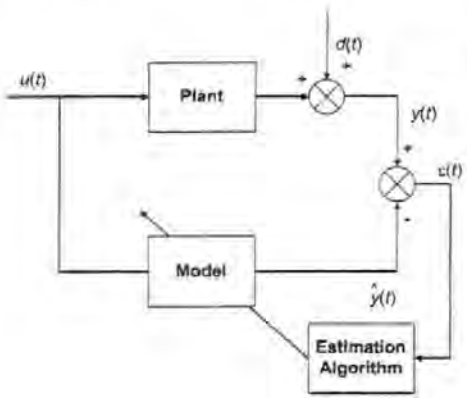


Fig. 2 The overall identification procedure

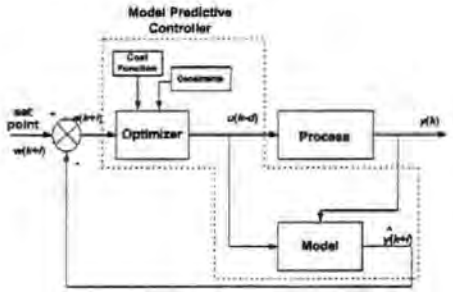


Fig. 3 Structure of a model predictive controller

respect to u. As a result the future tracking error is minimized. If there is no model mismatch, i.e. the model is identical with the process and there are no disturbances and constraints, the process will track the reference trajectory exactly on the sampling instants.

The MPC algorithm consists of the following three steps.

Step 1. Use a model explicitly to predict the process output along a future time horizon (prediction horizon).

Step 2. Calculate a control sequence along a future time horizon (control horizon H_u), to optimize a performance index.

Step 3. Employ a receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated as shown in Fig. 4.

The selection of MPC to control an AUV is attributed to several factors. Some of them are listed as follows:

1. The concept is equally applicable to single-input single-output systems and to multiple-input multiple-output systems.
2. MPC can be applied to linear and non-linear systems.
3. It can handle constraints in a systematic way during the controller design.
4. The controller is not fixed and is designed at every sampling instant based on actual sensor measurements so that disturbances can easily be dealt with compared with fixed-gain controllers.

In this paper, the performance index is optimized using a GA which is described next.

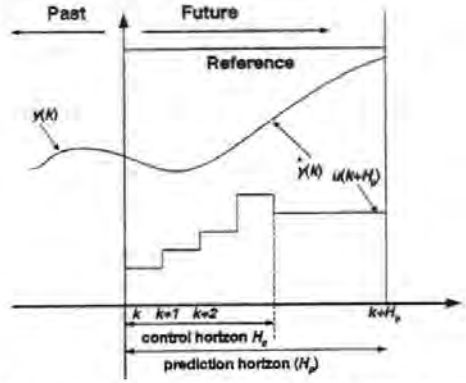


Fig. 4 Predicted output and the corresponding optimum input over a horizon H_p, where $u(k)$ is the optimum input, $\hat{y}(k)$ is the predicted output and $y(k)$ is the process output

4.1 Genetic algorithms

GAs inspired by Darwinian theory are powerful non-deterministic iterative search heuristics. GAs operate on a population consisting of encoded strings; each string represents a solution. The crossover operator is used on these strings to obtain new solutions, which inherits the good and bad properties of their parent solutions. Each solution has a fitness value; solutions having higher fitness values are most likely to survive for the next generation. The mutation operator is applied to produce new characteristics, which are not present in the parent solutions. The whole procedure is repeated until no further improvement is observed or the run time exceeds some threshold [41]. The flow chart of a simple GA is presented in Fig. 5 and the operation of the GA is explained as follows.

To start the optimization, GAs use randomly produced initial solutions. This method is preferred when *a priori* knowledge about the problem is not available. After randomly generating the initial population of, say, N solutions, the GA uses the three genetic operators to yield N new solutions at each iteration. In the selection operation, each solution of the current population is evaluated by its fitness normally represented by the value

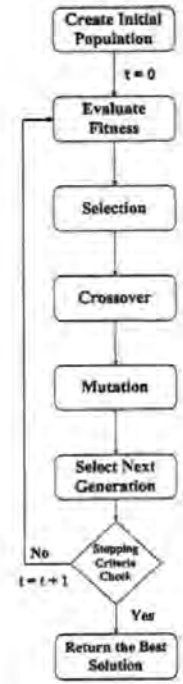


Fig. 5 Flow chart of a simple GA

of some objective function and individuals with higher fitness value are selected. Different selection methods such as roulette wheel selection and stochastic universal sampling can be used. The crossover operator works on pairs of selected solutions with a certain crossover rate. The crossover rate is defined as the probability of applying crossover to a pair of selected solutions. There are many ways of defining this operator such as single-point crossover, double-point crossover and multiple-point crossover. For example the single-point crossover works on a binary string by determining a point randomly in the two strings and corresponding bits are swapped to generate two new solutions.

Mutation is a random alteration with a small probability of the binary value of a string position. This operator prevents the GA from being trapped in local minima. The fitness evaluation unit in a GA acts as an interface between the GA and the optimization problem. Information generated by this unit about the quality of different solutions is used by the selection operation in the GA. Next the stopping criteria must be decided. This may be the case when there is no significant improvement in maximum fitness or the maximum allowable time (number of iterations) is passed. At the end of the algorithm, the best solution found so far is returned.

The key advantage of using GAs in an MPC framework is the ability to handle various forms of objective functions and process models without varying the controller structure. Both linear and non-linear models can be used with this approach without having to consider the local minima problem as is normally encountered in most linear optimization routines.

5 GA-BASED MPC ALGORITHM

The genetic-based control algorithm is depicted in Fig. 6. The algorithm was first proposed by Duwaish and Naeem [42] for the chemical processes identified as the Hammerstein and the Wiener models and resulted in a GA-based model predictive controller. As shown, the GA replaces the optimizer block and the AUV model identified from SI on the trials data has been used. The GA-based controller uses the process model to search for the control moves, which satisfy the process constraints and optimizes a cost function. The cost function to be minimized here is given by

$$J = \sum_{i=1}^{H_p} (k+i)^{\mathrm{T}} Q e(k+i) + \sum_{i=1}^{H_u} \Delta u(k+i)^{\mathrm{T}} R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^{\mathrm{T}} S u(k+i) \quad (2)$$

subject to

$$u^{\mathrm{l}} \leqslant u(k+i) \leqslant u^{\mathrm{u}}$$

$$\Delta u^{\mathrm{l}} \leqslant \Delta u(k+i) \leqslant \Delta u^{\mathrm{u}}$$

where the superscripts l and u represents the lower and upper bounds respectively. Q is the weight on the prediction error

$$e(k) = \hat{y}(k) - w(k) \quad (3)$$

where $w(k)$ is the reference or the desired set point. R and S are weights on the change in the input Δu and magnitude of the input u respectively. Adjusting the input weighting matrices could add damping to the closed-loop control system. The following steps describe the operation of the GA-based MPC algorithm and the flow chart of the algorithm is shown in Fig. 7.

Step 1. At time step k, evaluate process outputs using the process model.

Step 2. Use a GA search to find the optimal control moves that optimize the cost function and satisfy process constraints. This can be accomplished as follows:

(a) Generate a set of random possible control moves. The control moves or population consists of real values, which is reasonable in a real-world environment.

(b) Find the corresponding process outputs for all possible control moves using the process model.

(c) Evaluate the fitness of each solution using the cost function and the process constraints. The fitness function used here is given by

$$\text{Fitness} = \frac{1}{1+J} \quad (4)$$

where J is the cost function given by equation (2).

(d) Apply the genetic operators (selection, crossover and mutation) to produce a new generation of possible solutions. Roulette wheel selection and single-point crossover are used for selection of parents and mating respectively.

(e) Repeat until a predefined number of generations is reached and thus the optimal control moves are determined.

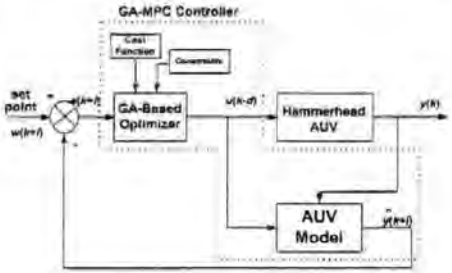


Fig. 6 GA-based model predictive controller

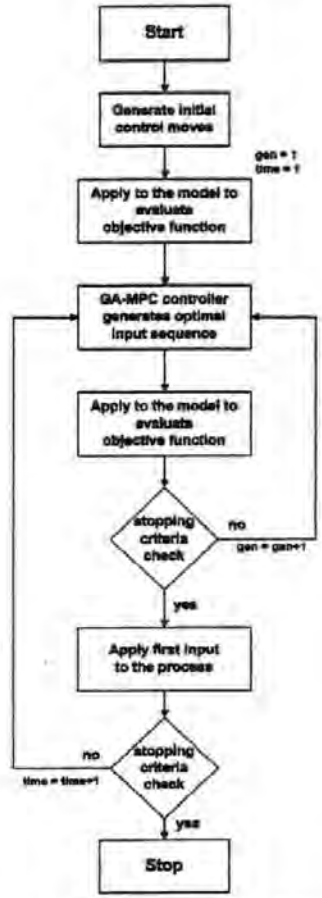


Fig. 7 Flow chart of the GA-based model predictive controller

Step 3. Apply the optimal control moves generated in step 2 to the process.

Step 4. Repeat steps 1 to 3 for time step $k+1$.

5.1 Constraints formulation

Constraints represent limitations on different physical quantities involved in a process. For instance, the input or output of a certain process is restricted beyond a specified value due to economical or environmental reasons or the input cannot be changed abruptly due to the hardware dynamics. One of the most powerful and distinguishing features of MPC is its ability to handle constraints in a natural way during the controller design at every sample time. Generally, two types of constraint are considered in controller design. Soft constraints are employed in the cost function as a penalty factor and can be violated to fulfil some other criteria. On the other hand, hard constraints represent physical limitations on actuators and cannot be violated. Here, hard constraints are placed on the input and rate of change in input (actuator bandwidth) variables. In this case, since the population in a GA represents the input variable, therefore, constraints are implemented by generating random initial population in the desired range:

$$u^{\mathrm{l}} \leqslant u \leqslant u^{\mathrm{u}}$$

Soft constraints on the input and rate of change in input are realized as penalty factors in the objective function in equation (2). To implement the hard rate constraints, the input for the current time instant k is compared with the input for the previous time instant $k-1$. If the difference Δu in the two inputs is violating the constraint, i.e. if it is higher or lower than the desired range, it is adjusted to the limiting value accordingly by manipulating the input at the current time instant k, thus not allowing the rate of change in inputs to violate the constraint. Mathematically, it can be stated as

$$u(k) = u(k-1) + \Delta u^{\mathrm{l}} \quad \text{if } u(k) - u(k-1) < \Delta u^{\mathrm{l}}$$

$$u(k) = u(k-1) + \Delta u^{\mathrm{u}} \quad \text{if } u(k) - u(k-1) > \Delta u^{\mathrm{u}}$$

6 EXPERIMENT DESIGN

The inputs applied to any system to be identified must be carefully designed prior to the experiment. This is imperative in obtaining good-quality data that contain sufficient information about the system dynamics. Reckless design of inputs could lead to useless data containing very little or no information about the system behaviour. Some common types of input sequence employed are uniformly distributed random numbers, pseudo-random binary sequence (PRBS) (Fig. 8a), and its variants such as multi-step (Fig. 8b) or doublet input (Fig. 8c).

Each of the inputs was applied for a specified duration during which data from the sensors were acquired. The following algorithm shows the order in which identification data were obtained.

Step 1. Send input to the control surface.

Step 2. While time < specified duration
read depth sensor
read TCM2 compass
read IMU
end.

Step 3. Read shaft speed encoder
read GPS
go to step 1.

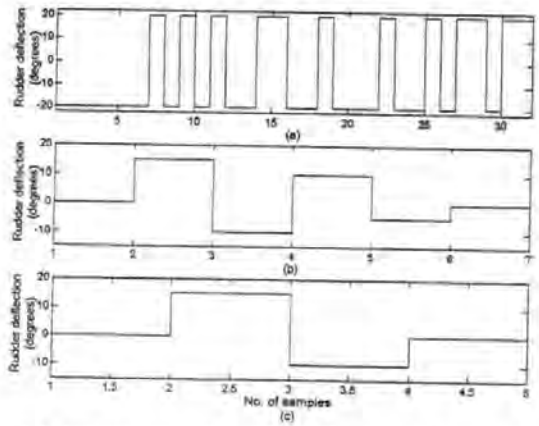


Fig. 8 Various inputs for SI: (a) PRBS; (b) multi-step; (c) doublet 38

With this configuration, the sampling frequency obtained was 8 Hz. The data were resampled afterwards at 1 Hz since this frequency was found to be adequate to control the *Hammerhead*. Moreover, it also helps to smooth the data, i.e. acts as a low-pass filter. It was observed that, during the transmission phase to the on-board actuators, no data could be acquired. This is due to the limited data acquisition capabilities of MATLAB. This problem was circumvented by leaving holes during that interval representing missing data. In addition, since there was no feedback from the control surfaces, the transition from one input to the other is approximated as a ramp and appropriate values are inserted. The whole input/output data was later processed and the missing data were interpolated. Figure 9a shows a data set with holes and Fig. 9b depicts the processed data.

The identification trials were carried out at Willen Lake, Milton Keynes, UK, while the controller was tested at Roadford Reservoir, Devon, UK. All experimental results shown in this paper were obtained with the vehicle manoeuvring on the surface at a fixed speed

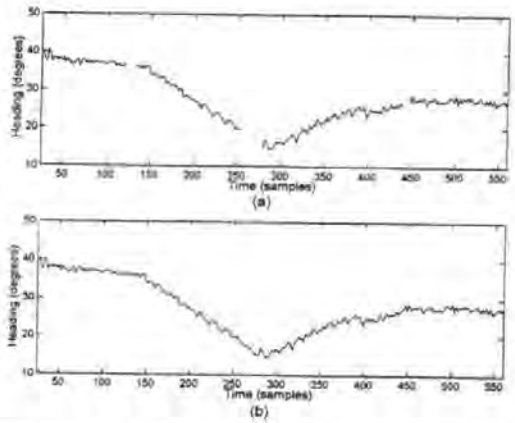


Fig. 9 *Hammerhead* yaw response to a doublet: (a) original data set; (b) interpolated data set

of approximately 2 knots. The acquired data were further processed to eliminate any transients and high-frequency noise. For the yaw channel identification, the heading data obtained from TCM2 and IMU were utilized only. However, the shaft speed was closely monitored and data at various speeds were carefully separated from the identification data.

An autopilot was then developed, simulated and tested in *Hammerhead*. The feedback for the controller was taken from the IMU as the IMU data are observed to be more accurate and reliable compared with the TCM2 data. For future trials, an MSDF algorithm [43] will provide an estimate of the true heading of the vehicle by combining the IMU and TCM2 outputs. It is important to point out the fact that the present authors were testing the controller on a 1960s-made vehicle hull with the bulk of the existing electronics, specifically the motors and their mechanical assembly, being retained. Because of this, the rudder movement could not be controlled precisely and the minimum deflection observed was 2°. For this reason, a minimum rate of change in input constraint was imposed on the rudder, which could lead to chattering effects in the rudder movement. The constraint implementation has been discussed in section 5.1.

7 RESULTS

This section presents results from the SI trials, model identification and controller development in a simulation environment followed by real-time experimental data. The controller code has been written in MATLAB and the model was identified using the SI toolbox in MATLAB.

7.1 *Hammerhead* modelling

One of the input sequences used in identification of the rudder yaw channel is a 32-length PRBS signal shown in Fig. 10a. The response of *Hammerhead* to this input is also illustrated in Fig. 10b. The data set is compared with the other heading responses with the same input magnitude and found to be consistent. An ARX(221) model was extracted from this data and is given by.

$$G(q) = \frac{-0.042\,26q^{-1} + 0.003\,435q^{-2}}{1 - 1.765q^{-1} + 0.7652q^{-2}} \tag{5}$$

where q^{-1} is the delay operator. This model has a pole very close to the unit circle which represents an integrator and exhibits a ramp output in response to a step input which is a characteristic of many AUVs. The autocorrelation and cross-correlation functions for this model are depicted in Fig. 11. Although the cross-correlation function was slightly outside the 95 per cent confidence band, the model was deemed adequate for further analysis. A cross-validation test is performed to gauge the predicting capability of the model. In this test, data not used for SI are applied to the model and the simulated output is compared with the measured output. Two cross-validation tests are performed for this channel and are shown in Figs 12 and 13. The simulated outputs as seen from the figures match the measured outputs reasonably well. However, there are some discrepancies evident from these figures due to the effect of surface currents on different data sets during the trials. From the MPC theory in section 4, it is clear that the basic controller performance relies heavily on an accurate model of the process and a robust controller should be able to cope with these discrepancies and disturbances.

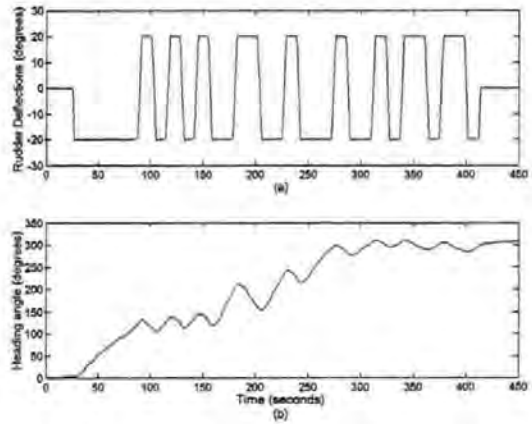


Fig. 10 *Hammerhead* yaw response to a 32-length PRBS input: (a) PRBS input; (b) yaw response from the IMU

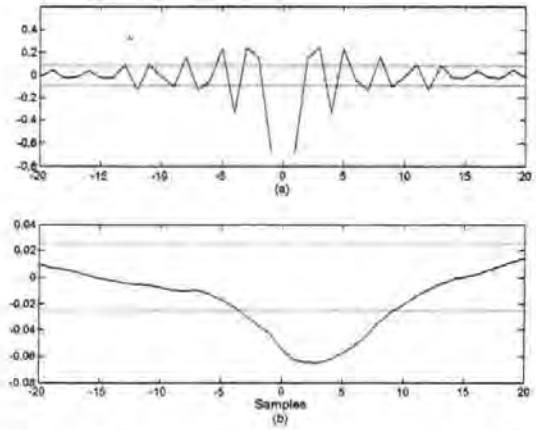


Fig. 11 (a) Autocorrelation of residuals for the yaw angle; (b) cross-correlation for the rudder angle and yaw angle residuals

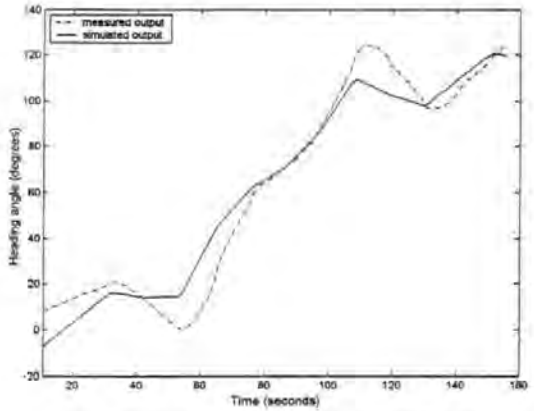


Fig. 12 First cross-validation test for the rudder yaw channel

The next section discusses the simulation and testing of GA-based MPC in the *Hammerhead* AUV.

7.2 GA-based model predictive controller results

To simulate the controller, it is assumed that there is no model–plant mismatch. The constraints on the rudder were $\pm 20°$ while the minimum allowed deflection is 2°, as discussed in the previous section. The weighting matrices were adjusted heuristically and the prediction and control horizons of MPC were set to minimize the control effort and to increase the speed of response. The tuning variables together with GA parameters used in the simulation are provided in Table 2.

The controller was simulated for a step change in heading and the result is depicted in Fig. 14. Without taking any disturbance into account, the vehicle seems to be following the desired course closely after initiating from an arbitrary direction. The response in Fig. 14b bears a small overshoot which can be minimized by adjusting the weighting matrices but at the cost of slow

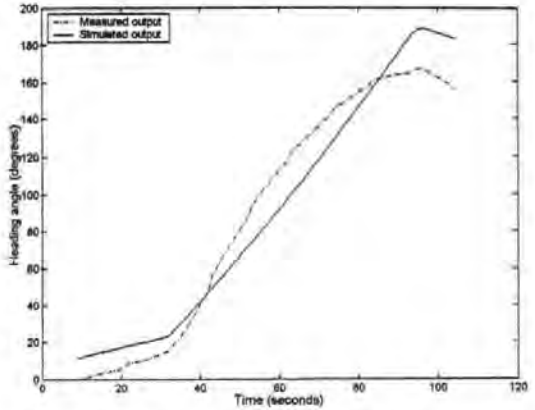


Fig. 13 Another cross-validation test for the rudder yaw channel

Table 2 Tuning parameters of the GA-based model controller

Parameter	Value
Q	1
R	0.5
S	0
H_p	7
H_u	1
Mutation probability	0.005
Crossover probability	0.7
Number of generations	10
Population size	100

response time. The rudder deflections generated by the GA-based controller are also shown in Fig. 14a requiring minimum control effort and stays within the specified bounds. There is a large movement in the rudder position around $t = 50$ s and yet this does not affect the vehicle's motion because of its slow dynamics. The spike is due to the probabilistic nature of GA which produces such results unless accounted for in the code but has not been implemented in this paper owing to the extra computational burden that it imposes. The chattering phenomenon can also be observed from this figure because of the 2° rate constraint. However, the effect of this is almost negligible on the vehicle's movement.

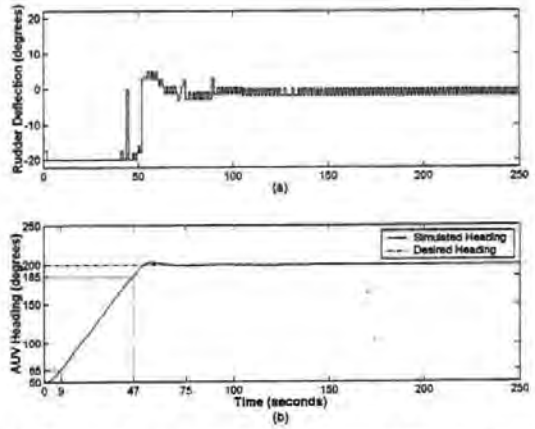


Fig. 14 GA-based MPC simulation results: (a) rudder deflections generated by the controller; (b) AUV heading

Next, the heading controller was tested in the *Hammerhead*. The parameters used in these trials were kept the same as in the simulation studies for a fair comparison. The test was carried out for a step change in heading. The initial and desired headings were kept the same and the vehicle was allowed to swim freely. It is evident from Fig. 15b that the GA-based model predictive controller was able to track the desired heading without any offset despite the presence of the always existing model uncertainty and external disturbances. However, there is an overshoot that could be blamed on the surface currents. Looking at the rudder deflections in Fig. 15a, again there is some expected chattering present. However, the rudder movements always remain within the specified constraints. There is a large spike followed by fluctuations in rudder movement at approximately 125 s in response to the change in vehicle's heading due to surface currents. However, the controller is robust enough to cope with it and attains the steady state input and output values in approximately 50 s. A statistical analysis reveals that the standard deviation of rudder deflections in experimental data is approximately 11°, while it is quite high (8°) in the simulation (even without any disturbance), possibly because of the chattering phenomenon.

It is interesting to note that the rise time in the experimental data is much smaller (24 s) than in the simulation (38 s). One reason for this is the effect of surface currents pushing the vehicle unwantingly and causing an even higher overshoot than the simulated response. Model–plant mismatch could also be another source of this problem.

8 CONCLUDING REMARKS

The problem of controlling underwater vehicles is a difficult task mainly because of lack of information regarding the vehicle dynamics. The paper reports some new results on the implementation of a GA-based model predictive controller on an AUV and suggests the use of SI on test trials data. Problems arising from experiments have been discussed together with some real-time experimental data. There are, however, several aspects which need to be investigated. The first and most-important is the effect of the umbilical on vehicle's movement. Although the present authors have tried to minimize its effect during the trials by conducting all tests on the water surface and by using as thin as possible tether, there is always some unavoidable drag present. This has deeply motivated the present authors to implement a wireless ethernet in the vehicle and to let the *Hammerhead* swim on its own during future trials.

Another potential source of argument is the use of an online GA. It has long been argued that the GA is not a potential solution for an online problem since it generates the solution of a problem from a set of random numbers based on probability. Additionally, the time complexity of a GA is more than the traditional optimization routines such as quadratic programming that makes it inadequate for fast-moving processes with short sampling times. However, considering the slow dynamics of *Hammerhead* (forward speed, approximately 2 knots) the present authors have successfully shown that MPC together with a GA can evolve into a powerful and robust online controller. The results showed the robust performance

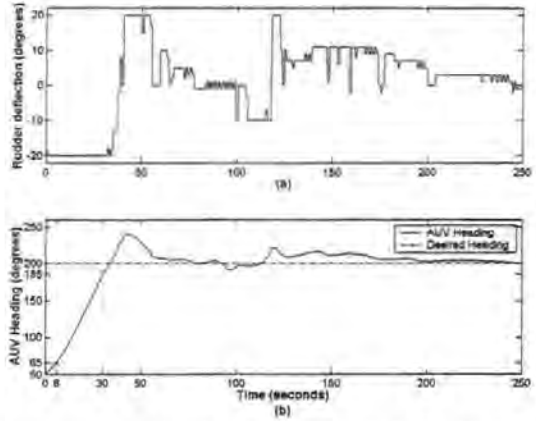


Fig. 15 Controller trial results: (a) rudder deflections generated by the GA-based model predictive controller; (b) *Hammerhead* heading obtained from the IMU

of the controller in the presence of model uncertainty and disturbances. Future trials will consist of the heading control at a certain depth with more precise knowledge on the vehicle position and orientation using fused sensors data. The present authors anticipate that the work presented herein will be considered as a valuable addition to the current AUV technology in the UK.

ACKNOWLEDGEMENTS

The authors would like to thank the Engineering and Physical Sciences Research Council for the funding of this project. Special thanks are due to the South West Water PLC for providing access to the Roadford Reservoir and to Willen Lake Management for their help in conducting the initial identification trials.

REFERENCES

1 **Lea, R. K.** Control of a tethered underwater flight vehicle. PhD thesis, University of Southampton, May 1998.
2 **Craven, P. J., Sutton, R.** and **Burns, R. S.** Control strategies for unmanned underwater vehicles. *J. Navig.*, 1998, **51**(1), 79–105.
3 **McPhail, S. D.** and **Pebody, M.** Navigation and control of an autonomous underwater vehicle using a distributed, networked, control architecture. *J. Soc. Underwater Technol.*, 1998, **23**(1), 19–30.
4 **Millard, N. W., Griffiths, G., Finegan, G., McPhail, S. D., Meldrum, D. T., Pebody, M., Perrett, J. R., Stevenson, P.** and **Webb, A. T.** Versatile autonomous submersibles—the realising and testing of a practical vehicle. *J. Soc. Underwater Technol.*, 1998, **23**(1), 7–17.
5 **Asakawa, K., Kojima, J., Ito, Y.** and **Takagi, S.** Autonomous underwater vehicle AQUA EXPLORER 1000 for inspection of underwater cables. In Proceedings of the 1996 Symposium on *Autonomous Underwater Vehicle Technology*, Monterey, California, USA, 1996, pp. 10–17 (IEEE, New York).
6 **Balasuriya, A.** and **Ura, T.** Autonomous target tracking by underwater robots based on vision. In Proceedings of the IEEE Conference on *Underwater Technology (UT'98)*, Tokyo, Japan, April 1998, pp. 191–197 (IEEE, New York).
7 **Ito, Y., Kato, N., Kojima, J.** and **Takagi, S.** Cable tracking for autonomous underwater vehicle. In Proceedings of the 1994 Symposium on *Autonomous Underwater Vehicle Technology*, Cambridge, Massachusetts, USA, 1994, pp. 218–224 (IEEE, New York).
8 **Kato, N., Ito, Y., Kojima, J., Takagi, S., Asakawa, K.** and **Shirasaki, Y.** Control performance of autonomous underwater vehicle 'AQUA EXPLORER 1000' for inspection of underwater cables. In Proceedings of the Conference on *Oceans Engineering for Today's Technology and Tomorrow's Preservation*, Brest, France, 1994, Vol. 1, pp. I/135–I/140 (IEEE, New York).
9 **Ishii, K., Fujii, T.** and **Ura, T.** An on-line adaptation method in a neural network based control system for auvs. *IEEE J. Oceanic Engng*, 1995, **20**(3), 221–228.
10 **Oliveira, P., Silvestre, C., Aguiar, A.** and **Pascoal, A.** Guidance and control of the SIRENE underwater vehicle: from system design to tests at sea. In Proceedings of the Conference on *Oceans*, Nice, France, September 1998, pp. 1043–1048.
11 **Aguiar, A.** and **Pascoal, A.** Modelling and control of an autonomous underwater shuttle for the transport of benthic laboratories. In Proceedings of the Conference on *Oceans*, Halifax, Canada, October 1997, pp. 888–895 (MTS–IEEE, New York).
12 **Healey, A. J.** and **Lienard, D.** Multivariable sliding model control for autonomous diving and steering of unmanned underwater vehicles. *IEEE J. Oceanic Engng*, July 1993, **18**(3), 327–339.
13 **Healey, A. J.** and **Marco, D. B.** Command, Control and Navigation: Experimental Results with the NPS ARIES AUV. *IEEE J. Oceanic Engineering*, October 2001, **26**(4), 466–477.
14 **Tinker, S. J., Bowman, A. R.** and **Booth, T. B.** Identifying submersible dynamics from free model experiments. *RINA An. Rep. Trans.*, 191–196, 1979.
15 **Ippoliti, C. G., Radicioni, S.** and **Rossolini, A.** Multiple models control of a remotely operated vehicle: Analysis of models structure and complexity. In Proceedings of the IFAC Conference on *Control Applications in Marine Systems (CAMS'2001)*, Glasgow, Scotland, 2001 (IFAC).
16 **Goheen, K. R.** and **Jefferys, E. R.** The application of alternative modelling techniques to ROV dynamics. In Proceedings of the IEEE International Conference on *Robotics and Automation*, Cincinnati, Ohio, USA, May 1990, pp. 1302–1309 (IEEE, New York).
17 **Ahmad, S. M.** and **Sutton, R.** Dynamic modelling of a remotely operated vehicle. In Proceedings of the 1st IFAC Workshop on the *Guidance and Control of Underwater Vehicles (GCUV 2003)*, Newport, South Wales, April 2003, pp. 47–52 (IFAC).
18 **Bossley, K. M., Brown, M.** and **Harris, C. J.** Neurofuzzy identification of an autonomous underwater vehicle. *Int. J. System Sci.*, 1999, **30**(9), 901–913.
19 **Kim, J., Kim, K., Choi, H. S., Seong, W.** and **Lee, K. Y.** Estimation of hydrodynamic coefficients for an AUV using nonlinear observers. *IEEE J. Oceanic Engng*, October 2002, **27**(4), 335–348.
20 **Kwiesielewicz, M., Piotrowski, W.** and **Sutton, R.** Predictive versus fuzzy control of autonomous underwater vehicle. In Proceedings of the IEEE International Conference on *Methods and Models in Automation and Robotics*, Miedzyzdroje, Poland, August 2001, pp. 609–612 (IEEE, New York).
21 **Naeem, W.** Model predictive control of an autonomous underwater vehicle. In Proceedings of the UKACC Conference on *Control*, September 2002, pp. 19–23 (UKACC, Sheffield).
22 **Naeem, W., Sutton, R.** and **Ahmad, S. M.** Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking (invited paper). Presented at the World Maritime Technology Conference, San Francisco, California, USA, 17–20 October 2003, Paper B1.

23 **Barrido, S. C.** and **Dadios, E. P.** Online robot tracking using genetic algorithms. In Proceedings of the IEEE International Symposium on *Intelligent Control*, Vancouver, Canada, October 2002, pp. 479–484 (IEEE, New York).

24 **Chou, W. D., Lin, F. J.** and **Shyu, K. K.** Incremental motion control of an induction motor servo drive via a genetic-algorithm-based sliding mode controller. *Proc. Instn Electl Engrs Part D: Control Theory Appl.*, May 2003, **150**(3), 209–220.

25 **Dalgleish, F. R., Tetlow, S.** and **Allwood, R. L.** The design of a laser-based imaging sensor for AUV navigation. In Institute of Physics PHOTON 02 Exhibition, Cardiff, Wales, September 2002.

26 **Dalgleish, F. R., Tetlow, S.** and **Allwood, R. L.** Preliminary experiments in the development of a laser-based imaging sensor for AUV navigation. In Proceedings of the Conference on *Guidance and Control of Underwater Vehicles (GCUV 2003)*, Newport, Wales, April 2003, pp. 239–244.

27 **Loebis, D., Dalgleish, F. R., Sutton, R., Tetlow, S., Chudley, J.** and **Allwood, R. L.** An integrated approach in the design of a navigation system for an AUV. In Proceedings of the Conference (MCMC 2003), Girona, Spain, September 2003, pp. 319–324.

28 **Ljung, L.** *System Identification, Theory for the User*, 2nd edition, 1999 (Prentice-Hall, Englewood Cliffs, New Jersey).

29 **Qin, S. J.** and **Badgewell, T. A.** An overview of nonlinear model predictive control applications. In *Nonlinear Model Predictive Control* (Edited by F. Allgöwer and A. Zheng), 1999 (Birkhäuser, Switzerland), pp. 369–392.

30 **Richalet, J., Rault, A., Testud, J. L.** and **Papon, J.** Model predictive heuristic control: Applications to industial processes. *Automatica*, 1978, **14**, 413–428.

31 **Cutler, C. R.** and **Ramaker, B. L.** Dynamic matrix control—a computer control algorithm. In Proceedings of the Joint Automatic Control Conference, San Francisco, California, USA, August 1980, Paper WP5-B.

32 **Clarke, D. W., Mohtadi, C.** and **Tuff, P. S.** Generalised predictive control. Part 1: the basic algorithm. *Automatica*, 1987, **23**(2), 137–148.

33 **Clarke, D. W., Mohtadi, C.** and **Tuff, P. S.** Generalised predictive control. Part 2: extensions and interpretations. *Automatica*, 1987, **23**(2), 149–160.

34 **Morari, M.** and **Lee J. M.** Model predictive control: past, present and future. *Computers Chem. Engng*, 1999, **23**, 667–682.

35 **Carlos, E. G., Prett, D. M.** and **Morari, M.** Model predictive control: theory and practice—a survey. *Automatica*, 1989, **25**(3), 335–348.

36 **Maciejowski, J. M.** *Predictive Control with Constraints*, 2002 (Prentice-Hall, Englewood Cliffs, New Jersey).

37 **Clarke, D.** (Ed.) *Advances in Model-Based Predictive Control*, 1994 (Oxford Science Publications, Oxford).

38 **Soeterboek, R.** *Predictive Control, A Unified Approach*, 1992 (Prentice-Hall, Englewood Cliffs, New Jersey).

39 **Richalet, J.** Industrial applications of model based predictive control. *Automatica*, 1993, **29**(5), 1251–1274.

40 **Rawlings, J. B.** Tutorial overview of model predictive control. *IEEE Control Systems Mag.*, June 2000, **20**(3), 38–52.

41 **Sait, S. M.** and **Youssef, H.** *Iterative Computer Algorithms with Applications in Engineering, Solving Combinatorial Optimization Problems*, 1999 (IEEE Computer Society, Los Alamitos, CA, USA).

42 **Duwaish, H.** and **Naeem, W.** Nonlinear model predictive control of Hammerstein and Wiener models using genetic algorithms. In Proceedings of the 2001 IEEE International Conference on *Control Applications (CCA01)*, Mexico City, Mexico, September 2001, pp. 465–469 (IEEE, New York).

43 **Loebis, D., Sutton, R., Chudley, J.** and **Naeem, W.** Adaptive tuning of a Kalman filter via fuzzy logic for an intelligent AUV navigation system. *Control Engng Practice*, 2004 (accepted for publication).

A simplified modular approach for the prediction of the roll motion due to the combined action of wind and waves

G Bulian and **A Francescutto***
Department of Naval Architecture, Ocean and Environmental Engineering, University of Trieste, Trieste, Italy

Abstract: In this work a combined analytical–numerical approach is proposed to address the problem of the ship roll motion under the combined action of wind and waves. Roll motion is modelled as a one-degree-of-freedom system non-linear in both damping and restoring. The approach is modular, allowing an easy update of the methodology on the basis of new research outcomes. Realistic environmental conditions regarding the effects of both wind and waves are taken into account and can be easily changed. The spatial correlation of wind gusts is taken into account by means of an 'aerodynamic admittance' function, whereas the moment due to waves is obtained from the sea slope spectrum using the concept of effective wave slope, leading to a 'hydrodynamic admittance' function. Both static and dynamic aspects of the problems are taken into account. The proposed analytical procedure, based on statistical linearization technique, allows approximate statistical averages of the roll motion, assumed to be Gaussian, to be obtained without necessarily resorting to time-consuming Monte Carlo simulations. On the basis of the results obtained, an estimation of the capsize probability can be carried out. It seems that the effect of wind gustiness could be considered very small when compared with the effects of waves and mean wind speed when the metacentric height is sufficiently large. Finally, the presented approach moves towards the concept of 'performance-based analysis', recently introduced at the International Maritime Organization as the basis for future developments of intact stability, in a clear and formal way.

Keywords: ship roll motion, wind, waves

1 INTRODUCTION: OVERVIEW OF THE METHOD AND BASIC ASSUMPTIONS

The present 'weather criterion' in the International Maritime Organization (IMO) intact stability code [1] is a rule intended to prevent extreme roll motions of the ship due to the combined effect of wind and waves. Although the idea is physically sound, many of the assumptions on which the criterion is based are questionable and are being criticized during the discussions for the revision of the 'intact stability code' at IMO [2]. The parameters used for the prediction of the rollback angle do not seem to be realistic for certain types of ship when not fitted with bilge keels or when fitted with bilge keels of limited size. Regarding the effect of wind, the gustiness factor assumed by present rules does not take into account the actual dimension of the ship and imposes a heeling moment under gust action which is 50 per cent larger than the mean heeling moment due to constant wind speed (assumed to be about 26 m/s). The environment assumed by the present criterion is basically deterministic, with some correction for taking into account its actual stochastic nature. Moreover, the methodology used by the criterion in the inherent estimation of the maximum roll angle after gust is not physical, because it assumes that, during the first swing after the gust, the effects of both waves and damping disappear, leading to an energy balance approach. The criterion is of the pass/fail type and no ranking is possible among different design proposals on the basis of this criterion.

Because of all this, the criterion is very difficult to modify in limited parts even using experimental results; this means that the criterion should be used as a 'black box', without possibilities for taking into account some important features of the ship under analysis. The determination of the 'level of safety' inherent in the criterion is also very difficult.

The MS was received on 15 April 2004 and was accepted after revision for publication on 21 June 2004.
* *Corresponding author: Department of Naval Architecture, Ocean and Environmental Engineering, University of Trieste, Via A Valerio 10, 34127 Trieste, Italy. email: francesc@units.it*

Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking

W Naeem, R Sutton and SM Ahmad,
Marine and Industrial Dynamic Analysis Research Group,
Department of Mechanical and Marine Engineering,
The University of Plymouth, Plymouth, UK

This paper investigates a new approach for the guidance and control of an autonomous underwater vehicle (AUV). An integrated system is developed and simulated involving a proportional navigation guidance (PNG) law and model predictive control (MPC). The classical PNG law for missile systems has been tailored to guide the AUV by generating reference headings. MPC is used to track the reference trajectory (guidance commands), which is optimised using a genetic algorithm (GA). The performance of the closed-loop system is evaluated in simulations with and without sea current disturbance and imposing actuator constraints. Simulation results for the case of a cable-tracking mission and waypoint following, clearly shows the superiority of the proposed algorithm.

AUTHORS' BIOGRAPHIES

Wasif Naeem obtained his Bachelor and Master degrees in electrical engineering from NED University of Engineering and Technology, Pakistan and King Fahd University of Petroleum and Minerals, Saudi Arabia, respectively. Currently he is pursuing a PhD programme in mechanical and marine engineering from the University of Plymouth, UK. He was employed as an R&D engineer from 1998 to 1999 in laser manufacturing, Pakistan, and was a research and teaching assistant during his MSc where he worked on a process control design project funded by SABIC industries.
Wasif Naeem's PhD is in the development and implementation of a guidance and control system of an autonomous underwater vehicle funded by EPSRC and in collaboration with Cranfield University, UK. His major interests are in the area of optimal control, model predictive control, system identification, process control and systems engineering.

Robert Sutton is professor of control systems engineering in the School of Engineering and head of the Marine and Industrial Dynamic Analysis Research Group at the University of Plymouth, UK. His main research interests lie in the application of advanced control engineering and artificial intelligence techniques to control problems. Prof Sutton's ongoing research is concerned with the use of fuzzy logic, neurofuzzy algorithms, artificial neural networks and adaptive search algorithms in the design of novel control systems for industrial and marine plant. He is the author of over 150 books, journal and refereed conference publications and on four occasions has been the recipient of premier awards from major engineering institutions for outstanding technical papers. He is a Fellow of the IMechE and the IEE.

Sarvat M Ahmad received his BEng in chemical engineering in 1992 from the University of Pune (India), and PhD from the department of Automatic Control and Systems Engineering, University of Sheffield, UK in May 2001. He worked as a Research Fellow at the University of Plymouth from April 2001 to Feb 2004, investigating modelling, guidance and control aspects of underwater vehicles. Currently, Dr Ahmad is employed by the University of Manchester as a research associate in the School of Engineering. His main research interests are: modelling, linear and non-linear system identification, and control of dynamical systems.

INTRODUCTION

The technology and applications of unmanned underwater vehicles (UUVs) have been improving at a rapid pace. From missions such as cable/pipeline inspection to oil exploration and to mine clearing operations, they are routinely being deployed by the offshore and defence industries. This is mainly attributed to the fact that it does not require a human onboard, thereby not jeopardising any life. In addition, in cases such as deep-sea exploration, where human intervention is not possible, they have proved to be a viable tool. Although regular monitoring and inspection of cables/pipelines running in deep sea have emerged as an important issue, little attention has been paid to sub-sea cables or pipelines. This paper describes a novel approach to underwater vehicle cable-tracking mission by employing an integrated guidance and control system using a PNG law for missile systems and MPC. The contemporary method to detect linear subsea objects is through active magnetic, passive magnetic or electromagnetic detectors mounted on a remotely operated vehicle (ROV).[1] These sensors provide lateral and longitudinal displacement of the ROV from the target pipeline, but no target direction. An additional sensor is needed to measure the target orientation. This information is then used by the ROV pilot to steer the vehicle over the pipeline. Although ROVs have been employed for detection and tracking, their range of operation is constrained by the length of the tether. Furthermore, the need for a support vessel and an ROV operator adds to the cost of monitoring operation. One way to circumvent these problems is to render the vehicle autonomous, ie, to execute the task with minimal human intervention.

A variety of methodologies and concepts have been reported to perform object-tracking by an underwater vehicle. An account of various AUV guidance schemes has recently been documented by Naeem et al[2] while a comparison of classical and advance control strategies has been reported by Craven et al.[3] In this paper, a modified PNG law is proposed for tracking underwater cables/pipelines employing a sonar system. MPC is used to track the reference commands generated by the PNG. The intent is to demonstrate the suitability of the integrated guidance and control scheme for detecting and tracking an undersea object – in this instance a pipeline – via simulation. The tracking of a pipeline by an AUV is first posed as an AUV-target interception problem. The classical PNG law is then employed to generate the guidance command signals to the AUV. Subsequently this is modified to achieve the desired target-tracking trajectory objective.

Sonars

Recent advances in sonar technology provide a sophisticated means of finding fibre optic cable, plastic, metal and other materials suspended in mid-ocean or buried in a seabed.[4] This strategy entails use of an active sonar system for target (pipeline) detection. Active sonars employ echo ranging to detect an object whereas passive sonars pick up acoustic radiation of ships, submarines etc, by an array of hydrophones. Some of the several other factors that influence this choice are:

1. *Active sonars* echo-range and therefore are capable of detecting even a submerged pipeline in the background of clutter ie, reverberations, in which it appears. Vision-based systems will have severe limitations in such a scenario which is very likely to occur at the seabed due to underwater currents and various other natural disturbances.
2. They can provide both range and orientation of the target, unlike magnetometers, which are non-directional and can easily mislead the AUV in presence of subsea ferrous deposits.
3. Presence of onboard *active sonar* can also be employed for retrieval of an AUV back to the mother ship once a mission is accomplished. This has been investigated by Ahmad et al[5] and is an area of ongoing research.
4. Sonic signals are the only practical and efficient way of long-range undersea communication, for instance between the mother ship and the AUV.[6]

The broader aim of the authors is to render an underwater vehicle truly autonomous, incorporating features such as smart launch, mid-course guidance, target tracking, area search and, finally, return and dock to the mother ship autonomously on completion of a given task.

PROBLEM DEFINITION

The following assumptions are made in order to formulate the guidance problem:

(i) The AUV-target engagement is planar, ie in the same plane.
(ii) Although the pipeline is a continuous object, it is convenient to assume it as a point mass moving with a constant velocity. This condition can be ensured by considering only the latest value of echoed 'ping' received by an onboard AUV sonar. The AUV is also considered as a constant-velocity point mass.
(iii) Complete navigational information of the target is available to the AUV.

Consider a two-dimensional engagement geometry in which the AUV and target are closing on each other at constant velocities V_p and V_t respectively, as shown in Fig 1. An imaginary line joining the AUV and target is referred as the line of sight (LOS). The angle formed by the

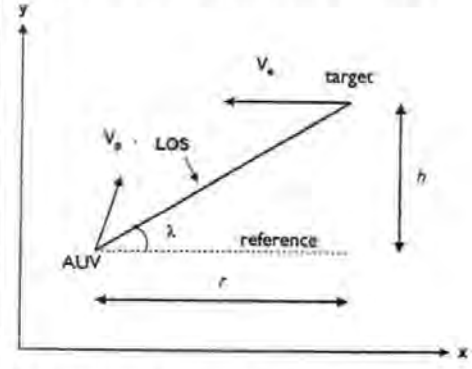


Fig 1: AUV-target engagement geometry

LOS with the fixed reference is λ and from the geometry is given as

$$\lambda = \tan^{-1}\frac{h}{r} \quad (1)$$

where, h and r are the relative separation between the AUV and target, perpendicular and parallel to the fixed reference respectively. The relative movement between the AUV and target causes the LOS to rotate through a small angle λ, indicating a displacement h between the AUV and target, perpendicular to the fixed reference. The length of LOS is a range R and represents the initial AUV–target distance. The problem is then to develop an integrated system which will make the initial range R between the AUV and target as small as possible at the end of expected intercept time. It will be shown later in simulation that it is a good starting point for achieving the desired tracking objective, without actually intercepting the target.

GUIDANCE AND CONTROL

Herein a PNG law is utilised to obtain the guidance commands. The guidance subsystem takes input from the sensors onboard the AUV. The sensors used could be global positioning system (GPS) for positioning on the surface, inertial navigation system (INS), compass etc. Information from the sensors is fused together and provided to the guidance system, which then generates commands to be followed by the AUV. A simple block diagram of the navigation, guidance and control system is depicted in Fig 2. MPC is used to track the reference commands from the guidance system. The selection of MPC for this paper is attributed to several factors, the most important being its ability to handle constraints in a natural and systematic way. The following subsections describe the PNG and MPC algorithms and their development.

Proportional navigation guidance law

The ultimate objective of the guidance law is to steer the AUV so that it will chase a target using a constant AUV velocity V_p and a controllable heading angle ψ_p. However, initially it will be regarded as an AUV-target interception problem and then subsequently modified to realise the desired 'tail-chase' type AUV trajectory. The tail-chase type trajectory of interest is akin to that formed when a dog is chasing a cat. This type of trajectory will ensure that the AUV is always trailing behind the target and thus continuously monitoring it at a close distance. From the above discussion of problem definition, it is intuitive that if the AUV is made to lie on the LOS and hold it there as well, a constant relative bearing between the AUV and target is ensured, that is, the LOS does not rotate, and interception will occur. This mechanisation can be realised using a PNG law.

Proportional navigation is a method of guidance, which generates command signals u_c, proportional to the LOS angle λ, so that the pursuing vehicle remains on the LOS. This can be mathematically stated as:

$$u_c \propto \lambda \quad (2)$$

$$u_c = k\lambda \quad (3)$$

where k is called the navigation constant and is an important design parameter.

A judicious choice of k will ensure that the LOS does not rotate and hence no further input command is required. Thus it influences both the engagement trajectory as well as the command input. The proportional navigation guidance scheme is illustrated in Fig 3 and a good description of PNG can be found in Garnel.[7]

Guidance law application

For implementing the guidance law of equation (3), it is necessary to compute the LOS angle λ. This requires relative positions of the AUV and target in both the co-ordinates, ie,

$$h = y_e - y_p \quad (4)$$

$$r = x_e - x_p \quad (5)$$

therefore,

$$\lambda = \tan^{-1}\left(\frac{y_e - y_p}{x_e - x_p}\right) \quad (6)$$

The components of the AUV velocity in the (x, y) plane can be stated as,

$$V_x = V_p \cos\psi_p \quad (7)$$

$$V_y = V_p \sin\psi_p \quad (8)$$

Hence, the differential equation for the components of the AUV position can be expressed as:

$$\dot{x}_p = V_x \quad (9)$$

$$\dot{y}_p = V_y \quad (10)$$

It is assumed that the AUV speed V_p and heading angle ψ_p are available to the guidance logic from an onboard speed log and gyro compass respectively. In certain cases both components of the AUV speed, ie equations (9) and (10) can be obtained directly from a Doppler log.

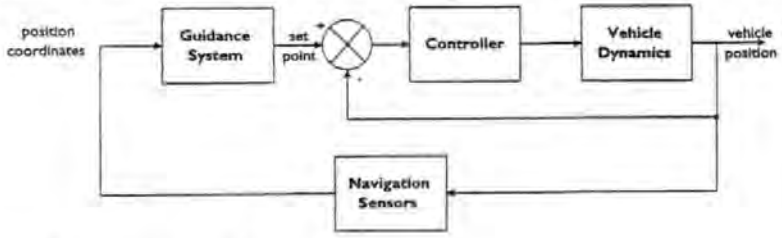


Fig 2: Navigation, guidance and control of a vehicle

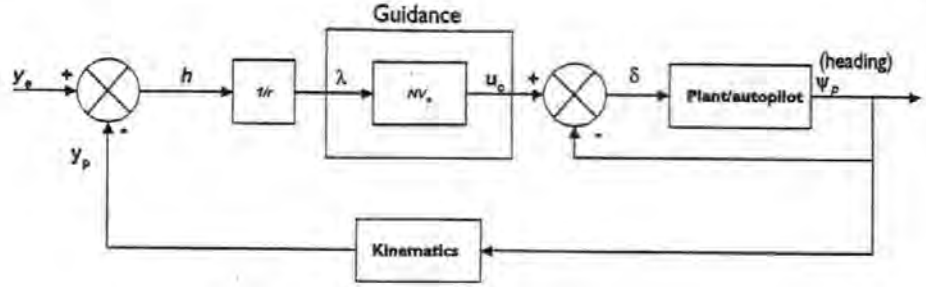


Fig 3: Proportional navigation guidance (PNG) Loop

By integrating the above velocity component equations, the AUV position components (x_p, y_p) in the earth-fixed coordinates can be found. Integrating equations (9) and (10) from time $t=0$ to $t=t_f$ and zero initial condition, ie $x_p(0)=0$ and $y_p(0)=0$ will give:

$$x_p = \int_0^{t_f} V_p \cos\psi_p \, dt \quad (11)$$

$$y_p = \int_0^{t_f} V_p \sin\psi_p \, dt \quad (12)$$

where t_f is time until intercept.

Similarly, it is easy to get the target positions (x_e, y_e) in the earth coordinates. It is assumed that the target velocity V_e and orientation ψ_e are known as a function of time. These quantities can be either measured or estimated. Therefore, target positions are given by

$$x_e = x_{e0} + \int_0^{t_f} V_e \cos\psi_e \, dt \quad (13)$$

$$y_e = y_{e0} + \int_0^{t_f} V_e \sin\psi_e \, dt \quad (14)$$

Thus, by substituting equations (11)–(14), in equation (6), the LOS angle λ can be determined which on substitution in equation (3) would generate appropriate guidance commands. This completes the guidance law mechanisation.

Model predictive control

Model predictive control (MPC) refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimise the future behaviour of a plant. Originally developed to meet the specialised control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace, metallurgy,[8] to name but a few.

The development of MPC can be traced back to 1978 after the publication of the paper by Richalet et al[9] called the *model predictive heuristic control* (MPHC). Then Cutler and Ramaker from Shell Oil in 1979 and 1980, developed their own independent MPC technology, *dynamic matrix control* (DMC).[10,11] However, the most popular form of predictive control, known as the *generalised predictive control* (GPC), has been devised by Clarke et al[12,13] and is employed in this paper.

The process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the output of the process can be used. Furthermore, if the process is subject to disturbances, a disturbance- or noise model can be added to the process model. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used. Typically, the criterion is the difference between the predicted process output and the desired reference trajectory. A simple criterion function is

$$J = \sum_{i=1}^{H_p} [\hat{y}(k+i) - w(k+i)]^2 \quad (15)$$

where $\hat{y}$ is the predicted process output, w is the reference trajectory, and H_p is the prediction horizon or output horizon. The structure of an MPC is shown in Fig 4. The controller output sequence u_{opt} over the prediction horizon is obtained by minimisation of J with respect to u. As a result, the future tracking error is minimised. If there is no model mismatch, ie the model is identical to the process and there are no disturbances and constraints, the process will track the reference trajectory exactly on the sampling instants. The following steps describe the MPC algorithm:

(i) Explicit use of a model to predict the process output along a future time horizon (prediction horizon).

(ii) Calculation of a control sequence along a future time horizon (control horizon), to optimise a performance index.

(iii) A receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated in Fig 5.

The selection of MPC to control an AUV is attributed to several factors. Some of them are listed below:

- The concept is equally applicable to single-input, single-output (SISO) as well as multi-input, multi-output systems (MIMO).

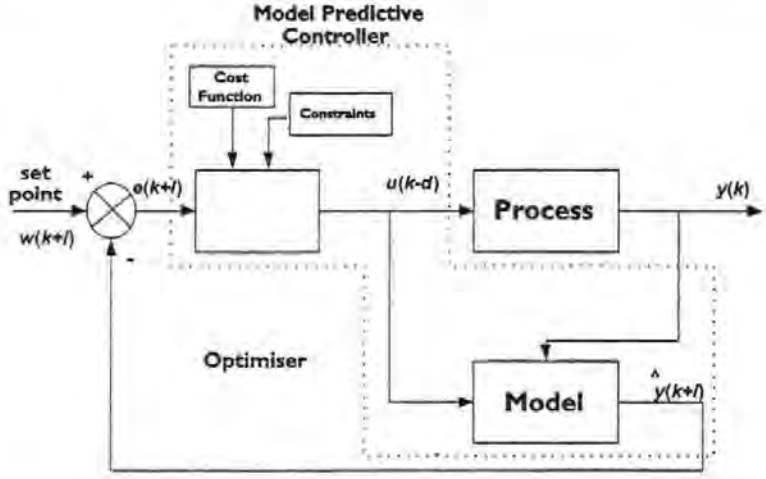


Fig 4: Structure of model predictive control (MPC)

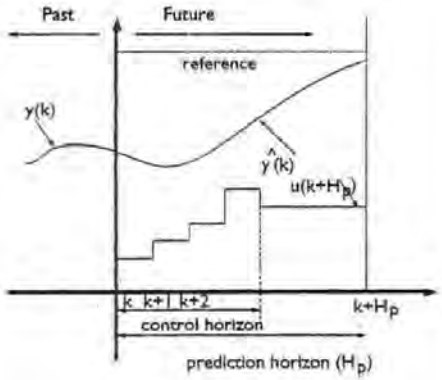


Fig 5: Predicted output and the corresponding optimum input over a horizon H_p, where $u(k)$, optimum input, $\hat{y}(k)$, predicted output, and $y(k)$, process output

- MPC can be applied to linear and non-linear systems.
- It can handle constraints in a systematic way during the controller design.
- The controller is not fixed, ie, it is designed at every sampling instant so disturbances can easily be dealt with.

The optimisation of the performance index is done using a GA, which is motivated by the work of Duwaish and Naeem.[14] The following section describes the operation of a simple genetic algorithm (GA).

Genetic algorithms

GAs, inspired by Darwinian theory, are powerful non-deterministic iterative search heuristics. GAs operate on a population consisting of encoded strings where each string represents a solution. A crossover operator is used on these strings to obtain the new solutions, which inherit the good and bad properties of their parent solutions. Each solution has a fitness value, and solutions having higher fitness values are most likely to survive for the next generation. A mutation operator is applied to produce new characteristics, which are not present in the parent solutions. The whole procedure is repeated until no further improvement is observed or run time exceeds to some threshold.[15] The flowchart of a simple GA is presented in Fig 6 and the operation is explained as follows.

To start the optimisation, GA uses randomly-produced initial solutions. This method is preferred when a priori knowledge about the problem is not available. After randomly generating the initial population of, say, N solutions, the GA uses the three genetic operators to yield N new solutions at each iteration. In the selection operation, each solution of the current population is evaluated by its fitness normally represented by the value of some objective function, and individuals with higher fitness value are selected. Different selection methods such as roulette wheel selection (RWS) and stochastic universal sampling (SUS) can be used. The crossover operator works on pairs of selected solutions with certain crossover rate where the crossover rate is defined as the probability of applying crossover to a pair of selected solutions. There are many ways of defining this operator, such as single point crossover, double point crossover, multipoint crossover etc. For example, the single point crossover works on a binary string by determining a point randomly in the two strings and corresponding bits are swapped to generate two new solutions.

Control law development

The MPC is responsible for directing the AUV towards the reference trajectory generated by the guidance system. In

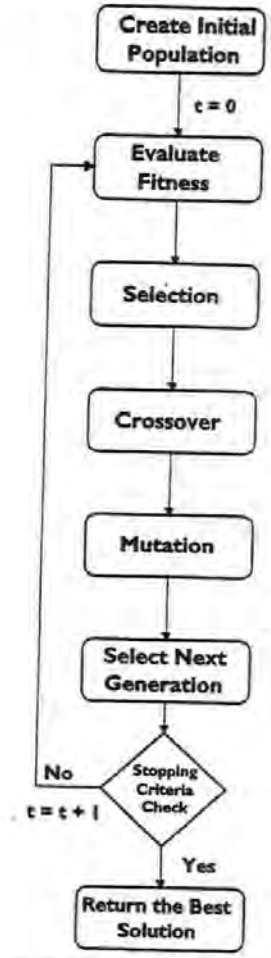


Fig 6: Flowchart of a simple genetic algorithm

order to generate the control moves, a cost function is minimised. The cost function used here is given by

$$J=\sum_{i=1}^{H_p} e(k+i)^T Q e(k+i)+\sum_{j=1}^{H_c} \Delta u(k+j)^T R \Delta u(k+j) \quad (16)$$

subject to

$$u^l \leq u(k+j) \leq u^u$$

where the superscripts l and u represents the lower and upper bounds on the input moves respectively. R is the weight on the rate of change of control moves and Q is the weight on the prediction error

$$e(k)=\hat{y}(k)-w(k)$$

where $\hat{y}(k)$ is the predicted process output and $w(k)$ is the reference trajectory generated by the PNG law. The second term in equation (16) represents the penalty on the rate of change of control moves. This is augmented to prevent excessive movements of the rudder.

The following steps describe the operation of the MPC algorithm using GA. At any sample time k:

1) Evaluate process outputs using the process model.
2) Use GA search to find the optimal control moves which optimise the cost function and satisfies process constraints. This can be accomplished as follows.
   (a) Generate a set of random possible control moves.
   (b) Find the corresponding process outputs for all possible control moves using the process model.
   (c) Evaluate the fitness of each solution using the cost function and the process constraints.
   (d) Apply the genetic operators (selection, crossover and mutation) to produce a new generation of possible solutions.
   (e) Repeat until the predefined number of generations has been reached and thus the optimal control moves are determined.
3) Apply the first optimal control move of the sequence generated in step (2) to the process.

SIMULATION RESULTS

The proposed guidance and control algorithm has been applied to an AUV simulation model supplied by QinetiQ, based on the AUTOSUB vehicle,[16] having a torpedo-shaped hull. Dimensionally, the model represents an AUV that is 7m long, approximately 1m in diameter and has a nominal displacement of 3600kg.

The equations of motion describing the dynamic behaviour of the vehicle in the yaw, sway and roll modes can be written in the state-space representation as given by Marshfield:[17]

$$\mathbf{E}\dot{\mathbf{x}}=\mathbf{F}\mathbf{x}+\mathbf{G}\mathbf{u} \quad (17)$$

where

$$\mathbf{E}=\begin{bmatrix} (m-Y_{\dot{v}}) & -Y_{\dot{R}} & 0 & -(Y_{\dot{p}}+mZ_G) & 0 \\ -N_{\dot{v}} & (I_Z-N_{\dot{R}}) & 0 & -N_{\dot{p}} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -(K_{\dot{v}}+mZ_G) & -K_{\dot{R}} & 0 & (I_X-K_{\dot{p}}) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{F}=\begin{bmatrix} Y_{UV}U & (Y_{UR}-m)U & 0 & Y_{UP}U & 0 \\ N_{UV}U & N_{UR}U & 0 & N_{UP}U & 0 \\ 0 & 1 & 0 & 0 & 0 \\ K_{UV}U & (K_{UR}+mZ_G)U & 0 & K_{UP}U & -mgBG \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$G=\begin{bmatrix} Y_{UU\delta ru}U^2 & Y_{UU\delta rl}U^2 & 0 & 0 & 1 & 1 \\ N_{UU\delta ru}U^2 & N_{UU\delta rl}U^2 & l_\phi & -l_\phi & l_\psi & -l_\psi \\ 0 & 0 & 0 & 0 & 0 & 0 \\ K_{UU\delta ru}U^2 & K_{UU\delta rl}U^2 & \gamma_\phi & -\gamma_\phi & \gamma_\psi & -\gamma_\psi \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$u=[\delta_{Stern-upper}\ \delta_{Stern-lower}\ \delta_{Stern-port}\ \delta_{Stern-starboard}\ \delta_{Bow-upper}\ \delta_{Bow-lower}]^T$$

and the state variables are v, r, ψ, p, ϕ (see Appendix A for a nomenclature). However, it should be noted that, for this study, the upper and lower canards are the only surfaces used to control the yaw dynamics. A simplified linear model of the yaw dynamics of the vehicle is extracted from the above set of equations using system identification techniques. The identified model is of the form:

$$\dot{x}=Ax+Bu \tag{18}$$

where, A and B are the state and input coupling matrices respectively. More precisely, the two dimensional core state-space model is given by,

$$\begin{bmatrix}\dot{\psi}\\ \dot{r}\end{bmatrix}=\begin{bmatrix}a & b\\ c & d\end{bmatrix}\begin{bmatrix}\psi\\ r\end{bmatrix}+\begin{bmatrix}e\\ f\end{bmatrix}u \tag{19}$$

where a, b, c, d, e, and f are constant parameters. Moreover, the continuous-time yaw model of the AUV is discretised at a sampling frequency of 2Hz owing to the requirements of the digital MPC controller.

The ultimate objective of this paper is the development and simulation of an integrated guidance and control system for an AUV to follow a subsea cable/pipeline for inspection purpose. The program has been written in MATLAB/SIMULINK environment. The guidance law is developed in SIMULINK while the control system has been designed in MATLAB which are then combined to form an integrated guidance and control system as shown in Fig 7. It was mentioned earlier, in the section on problem definition, that the AUV and pipeline are considered as point masses moving at a constant velocity. It is also assumed that the AUV and target are moving at the same speed, ie,

$$\frac{V_p}{V_e}=1 \tag{20}$$

The target frame of coordinate (FOC) with respect to the AUV is (0,10) representing the seabed, whereas, the initial AUV coordinates in the inertial two-dimensional frame of reference (x,y) plane are (0, 200) with respect to the target FOC. Further, it is assumed that the target obeys equation (21), and is heading eastwards from the initial FOC.

$$\left.\begin{aligned} x(t)&=x_0+V_e t\\ y(t)&=h \end{aligned}\right\} \tag{21}$$

V_e and h being fixed. The target, a 'fleeing' pipeline, is travelling at a constant distance h from the AUV's inertial FOC. The AUV is to be launched from a mother ship in the vicinity (0, 200) of the target to intercept it. This completes the tail-chase problem definition. A navigation constant of

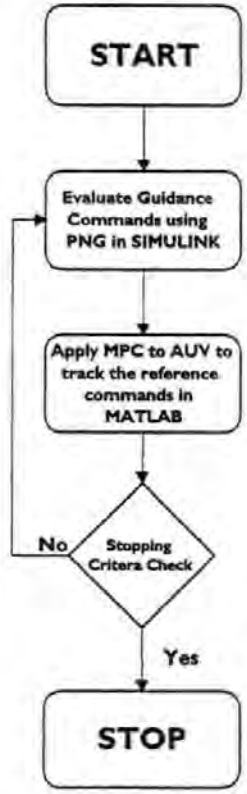


Fig 7: Integration of guidance and control systems in MATLAB/SIMULINK environment

$k=1$ has been chosen since, for this value, the AUV trajectory changes at the same rate as the imaginary LOS joining the target and the AUV. This type of flight profile is often referred as 'pursuit course' and the corresponding guidance law as 'pure pursuit'. The trajectory is similar to that formed by a predator when pursuing a prey, for instance, a dog-cat or hound-hare pursuit. The predator always prefers to tail chase a target rather than intercept it by establishing a lead angle. This characteristic is exploited by the authors to achieve the pipeline-tracking objective and is discussed next.

Since it is desired to *follow* rather than *intercept* the target, a bias is introduced to equation (4), which in effect alters the guidance signal issued by the PNG law of equation (3). This essentially prevents the value of h in equation (4) from reducing to zero, thus precluding the AUV from intercepting the target. The value and the time of introduction of the bias would be user-defined, depending on at what depth above the target (pipeline) the AUV is expected to operate.

Table 1: Simulation parameters for the GA and MPC

Parameter	Value
Prediction horizon	
Control horizon	
Population size	
Number of generations	
Mutation probability	
Crossover probability	
Q	
R	

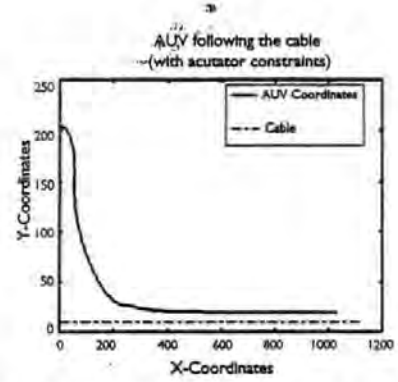


Fig 8: AUV and target position coordinates for the constrained case. AUV is tracking the cable at a specified height

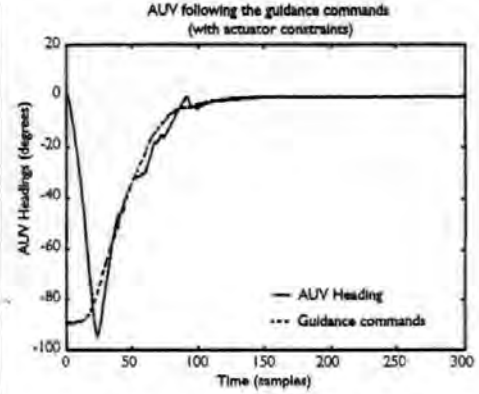


Fig 9: AUV heading controlled by the MPC following closely the guidance commands generated by the PNG

Parameters used in all the simulations are provided in Table 1. The actuator movement is constrained between ±25deg. A bias of 10m is introduced after 40 sample times of the simulation run. In a real system, this value could be kicked-off by a pressure-depth sensor on an AUV, after descending to a depth of 10m above the seabed. The simulation is run for 300 samples and the result is depicted in Fig 8. The AUV heading and rudder deflections are also shown in Figs 9 and 10 respectively. The AUV charts out a 'pursuit course' for the first 150m of distance travelled. With the introduction of a 10m bias signal at the end of 150m, the vehicle maintains a desired longitudinal position h, while tracking the cable laterally without intercepting it.

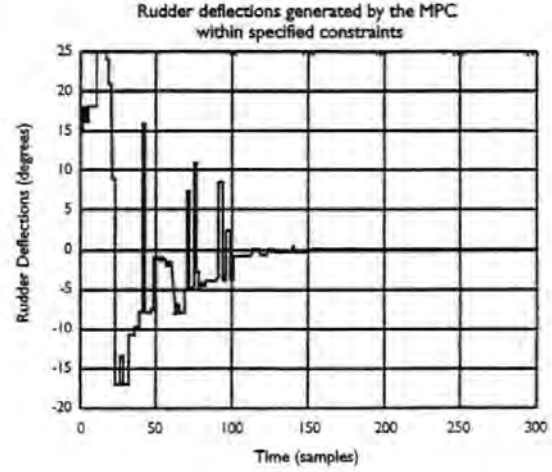


Fig 10: Rudder deflections generated by the controller within specified constraints needed to track the reference trajectory (guidance commands)

Next, waypoint-following by Healey and Lineard[18] is considered and the effect of sea current disturbance is investigated. The circle of acceptance is taken as twice the length of the vehicle. First, simulation is run without any sea current disturbance with actuator movement restricted to 25deg in either left or right direction. The result is depicted in Fig 11 clearly showing that the AUV is closely following the waypoints without much control effort and within the constrained limit as shown in Fig 12. The mean square error (MSE) between the actual and ideal AUV distances from the waypoints without any disturbance is approximately 45m^2. Finally, the simulation is run for a sea current disturbance of 1ms^{-1} in the positive y-direction and with actuator constraints. As shown in Figure 13, the disturbance is trying to knock the vehicle off the track, but the controller is still able to cope with it and reaches the target waypoints. The MSE is about 800m^2, much larger than the no disturbance case, although the vehicle follows all the waypoints even in the worst case scenario of the sea current mentioned above. The rudder movement is also confined within the specified limits as depicted in Fig 14, although it is lot more aggressive as compared to the no disturbance case. A statistical analysis reveals that the standard deviation of rudder deflections in the presence of wave disturbance is about 10deg while it is approximately half that value without any disturbance.

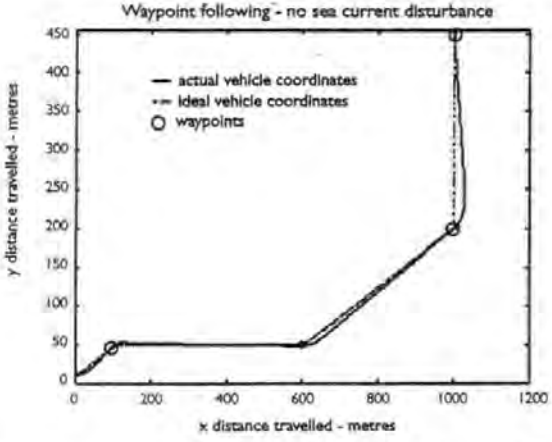


Fig 11: AUV and target position coordinates for the constrained case without sea current disturbance. The AUV is closely following the target waypoints

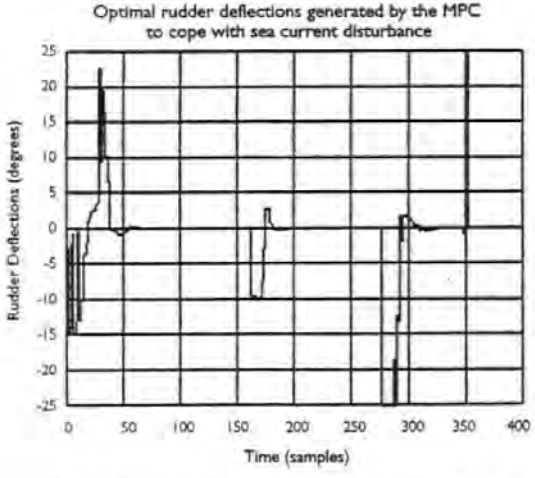


Fig 12: Rudder deflections generated by the controller needed to track the waypoints for the constrained case and without sea current disturbance

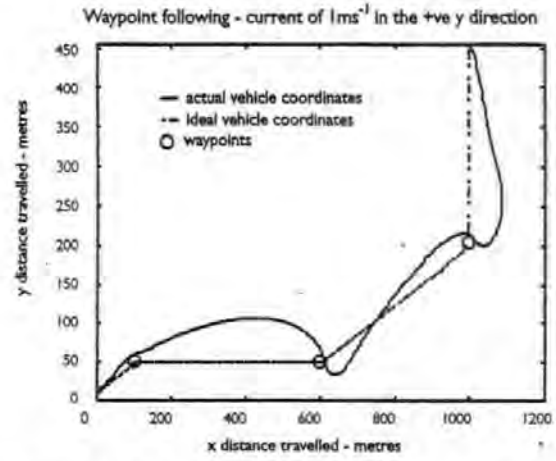


Fig 13: AUV and target position coordinates for the constrained case with sea current disturbance in the positive y-direction. The AUV is closely following the target waypoints

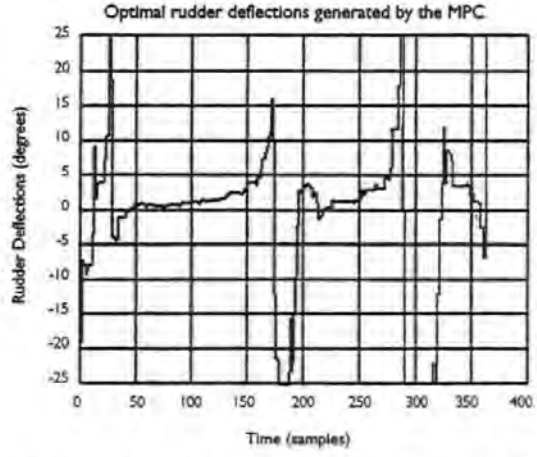


Fig 14: Rudder deflections generated by the controller needed to track the waypoints for the constrained case and with sea current disturbance in the positive y-direction

CONCLUSIONS

Periodic inspection and monitoring of sub-sea cables/pipelines have emerged as important issues. These operations are currently being performed using ROVs whose endurance and capabilities are limited.

This paper proposes sonar-based detection and tracking of ocean floor pipelines and cables and is posed as a two-stage problem. In the first stage, utilisation of classical PNG law is demonstrated in intercepting a *fleeing target*, while in the second stage the guidance law is modified to achieve the target-tracking objective, albeit without ever intercepting it. MPC is used to generate control commands for the actuators to keep the vehicle as close as possible to the reference trajectory. Actuator constraints are also handled in an efficient way during the controller design. The proposed integrated guidance and control system accomplishes

target detection as well as tracking objective without losing sight of the target. Incorporation of this guidance and control scheme is expected to increase the range of AUV, as no human intervention is essential for guiding the vehicle. The range would only be limited by the availability of the onboard power supply.

REFERENCES

1. Bjerrum A, and Slater T. 2001. *Autonomous tracking of submarine pipelines and cables*. Hydrographic Society, Proceedings of Hydro 2001, March, Norwich, UK.

2. Naeem W, Sutton R, Ahmad SM, and Burns RS. 2003. *A review of guidance laws applicable to unmanned underwater vehicles*. The Journal of Navigation, 56(1): pp. 15-29.

3. Craven PJ, Sutton R, and Burns RS. 1998. *Control strategies for Unmanned Underwater Vehicles*. The Journal of Navigation, 51(1): pp. 79-105.

4. Bannon RT. 1998. *ROVs and undersea cable maintenance*. In: Proceedings of the Underwater Technology '98, April, Tokyo, Japan.

5. Ahmad SM, Sutton R, and Burns RS. *Retrieval of an autonomous underwater vehicle: An interception approach*. Submitted to the Journal of Underwater Technology.

6. Whitcomb LL. 2000. *Underwater robotics: out of the research laboratory and into the field*. IEEE International Conference on Robotics and Automation, USA.

7. Garnell P. 1980. *Guided Weapon Control Systems*, 2nd ed., Brassey's defence publishers.

8. Qin SJ, and Badgewell TA. 1997. *An overview of industrial model predictive control technology*. http://www.che.utexas.edu/~qin/ps/cpcv16.ps

9. Richalet J, Testud J, Rault A, and Papon J. 1978. *Model predictive heuristic control: Applications to industrial processes*. Automatica, vol. 14, pp. 413-428.

10. Cutler C, and Ramaker B. 1979. *Dynamic matrix control—a computer control algorithm*, AIChE National Meeting, Houston, TX, USA.

11. Cutler C, and Ramaker B. 1980. *Dynamic matrix control, a computer control algorithm*. In: Proceedings of the Joint Automatic Control Conference, Paper WP5-B, San Francisco, CA, USA.

12. Clarke DW, Mohtadi C, and Tuff PS. 1987(a). *Generalised predictive control. Part 1: The basic algorithm*. Automatica, 23(2): pp. 137-148.

13. Clarke DW, Mohtadi C, Tuff PS. 1987(b). *Generalised predictive control. Part 2: Extensions and Interpretations*. Automatica, 23(2): pp. 149-160.

14. Duwaish H, and Naeem W. 2001. *Nonlinear model predictive control of Hammerstein and Wiener Models using Genetic Algorithms*. In: Proceedings of the 2001 IEEE International Conference on Control Applications (CCA'01), 5-7 September, Mexico City, Mexico, pp. 465-469, IEEE.

15. Sait SM, and Youssef H. 1999. *Iterative computer algorithms with applications in engineering, solving combinatorial optimisation problems*. IEEE Computer Society Press and John Wiley & Sons, Inc.

16. Millard NW, Griffiths G, Finegan G, McPhail SD, Meldrum DT, Pebody M, Perrett JR, Stevenson P, and Webb AT. 1998. *Versatile Autonomous Submersibles – the realising and testing of a practical vehicle*. Underwater Technology, 23(1): pp. 7-17.

17. Marshfield WB. 1992. *Submarine Data Set for use in Autopilot Research*. Technical Memorandum, DRA/MAR TM (MTH) 92314, DRA Haslar, April.

18. Healey AJ, and Lineard D. 1993. *Multivariable Sliding Model Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles*. *IEEE Journal of Oceanic Engineering*, 18(3): pp. 327-339, July.

APPENDIX

Nomenclature of the AUV equation parameters

E, F, G	State equation matrices
m	Mass
p, r	Roll and yaw angular velocity components
u, v	Surge and sway linear velocity components
ψ, ϕ	Yaw and roll angles
I_X, I_Z	Inertia components
K, N	Roll and yaw moments
Y	Y direction force component
B	Buoyancy force
G	Centre of mass
K_{UP}	Dimensional hydrodynamic coefficients of roll
$N_{UU\delta rv}$	Dimensional hydrodynamic coefficient of yaw with respect to canard upper
$\lambda\phi$, λ_ψ, γ_ϕ, γ_ψ	Roll and yaw moment arm lengths
$\delta_{Bow\text{-}upper}$	Input from upper canard rudder
$\delta_{Stern\text{-}port}$	Input from port stern hydroplane
$\delta_{Stern\text{-}upper}$	Input from upper stern rudder

THE JOURNAL OF NAVIGATION (2003), 56, 15–29. © The Royal Institute of Navigation
DOI: 10.1017/S0373463302002138 Printed in the United Kingdom

A Review of Guidance Laws Applicable to Unmanned Underwater Vehicles

W. Naeem, R. Sutton, S. M. Ahmad and R. S. Burns

(The University of Plymouth)

The main problem in bringing autonomy to any vehicle lies in the design of a suitable guidance law. For truly autonomous operation, the vehicle needs to have a reliable Navigation, Guidance and Control (NGC) system of which the guidance system is the key element that generates suitable trajectories to be followed. In this review paper, various guidance laws found in the literature and their relevance to autonomous underwater vehicles (AUVs) are discussed. Since existing guidance laws for underwater vehicles have emulated from tactical airborne missile systems, a number of approaches for the missile guidance systems are considered. Finally, potential guidance strategies for AUVs are proposed.

KEY WORDS

1. Guidance. 2. Command & Control. 3. Underwater Vehicles.

1. INTRODUCTION. In recent years, control systems have assumed an increasingly important role in the development and advancement of modern civilisation and technology. In particular, the burgeoning in the field of navigation, guidance and control (NGC) systems, spurred on mainly by the challenges of unsolved aerospace problems, contributed significantly to progress achieved in the development of modern systems and control theories. The success of the Soviet Union's satellite technology in the 1950s stimulated the United States to develop their own aerospace technology thus creating between the two of them new concepts in the field of control system design. The Apollo programme in the 1960s is a classical example of the translation of various NGC concepts into working systems. The early success of NGC systems soon led to advances in such diverse areas as industrial manufacturing, energy management (Lin, 1991) and underwater vehicles. Although applications of NGC in these areas have shown a profound impact in control theory in general, the majority of research and development in NGC continues to find its main application in the aerospace industry.

Navigation, guidance and control of airborne systems have been reported extensively in the literature (Cloutier *et al.*, 1989; Lin and Su, 2000; Lin, 1991); however, little attention has been paid to the issue of guidance of autonomous underwater vehicles (AUVs). In light of this, the impetus behind this paper is to review a number of approaches that have been adopted for the guidance of air and sea vehicles with an emphasis on AUVs. Furthermore, it is the intention to explore ways and means of employing successful guidance strategies of air-based systems to underwater vehicles.

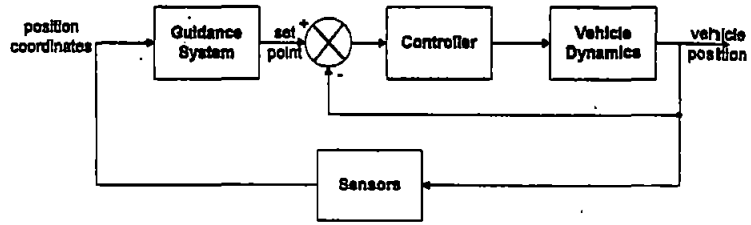


Figure 1. Guidance and Control for a Vehicle.

This would also entail certain modifications to suit the underwater mission requirement.

The paper is organised as follows. Section 2 explores the guidance problem in general. Guidance laws for the airborne missile systems are presented in Section 3, while Section 4 discusses AUVs and the guidance laws associated with them. Some modified guidance laws for underwater cable tracking problem are proposed in Section 5. Finally, concluding remarks are given in Section 6.

2. GUIDANCE. All autonomous vehicles must have on-board NGC systems, which should work in accord with each other for proper operation. Imperfections in one system degrade the efficiency of the others. The navigation system provides information related to the target, which is processed by the guidance system to generate reference headings. The control system is responsible for keeping the vehicle on course as specified by the guidance processor. In remotely operated systems, guidance commands are sent from a ground station, while autonomous vehicles have an on-board guidance processor. In this respect, a guidance system plays the vital role in bringing autonomy to the system. Some definitions and a brief description of the elements of a guidance system are presented as follows.

> Guidance is the action of determining the course, attitude and speed of the vehicle, relative to some reference frame, to be followed by the vehicle (Fossen, 1994).

From the perspective of a control system:

> guidance is a matter of finding the appropriate compensation network to place in series with the plant in order to accomplish an intercept (Lin, 1991).

The guidance system decides the best trajectory (physical action) to be followed by a vehicle based on target location and vehicle capability.

The primary function of the elements that constitute a guidance system are sensing, information processing and correction. A rudimentary guidance and control system for a vehicle is shown in Figure 1. As shown, the guidance system receives inputs from all the sensors on-board and generates the relevant signals or set points for the control system. Guidance issues are mainly determined by the nature and location of the target and the environmental conditions. The nature of the target corresponds to the condition as to whether or not the target is stationary, moving, or manoeuvring. The target location is also imperative as it determines the heading to be followed by the vehicle; however, the accuracy of the system depends on the environmental conditions. The

guidance problem is also related closely to the bandwidth of the system. It is often assumed while formulating the problem that the controller has a sufficiently large bandwidth to track the commands from the guidance subsystem (Sutton *et al.*, 2000); however, in practice, true vehicle capability can only be measured in the presence of constraints such as system dynamics and actuator limitations.

The definitions and elements of a guidance loop discussed above are quite generic and refer to all guidance mechanisms. Although widely employed in the aerospace and land vehicles, they are equally valid for underwater vehicles.

3. MISSILE GUIDANCE.

The guidance technology of missiles is a mature field with an abundance of guidance laws already implemented in real systems. Many different guidance laws have been employed exploiting various design concepts over the years. Currently, the popular terminal guidance laws involve line-of-sight (LOS) guidance, LOS rate guidance, command-to-line-of-sight (CLOS) guidance, proportional navigation (PNG) (Locke, 1955), augmented proportional navigation guidance (APNG) (Zarchan, 1994) and optimal guidance laws based on linear quadratic regulator theory (Bryson and Ho, 1969; Nazaroff, 1976), linear quadratic Gaussian theory (Potter, 1964) or linear quadratic exponential Gaussian theory (Speyer *et al.*, 1982).

Among the current techniques, guidance commands proportional to the LOS angle rate are generally used by most high-speed missiles today to correct the missile course in the guidance loop. Recently, many advance strategies have been implemented to generate different guidance laws. Rajasekhar *et al.* (2000) uses fuzzy logic to implement PNG law. The fuzzy law generates acceleration commands for the missile using closing velocity and LOS rate as input variables. The input data is fuzzified and their degree of membership to the output fuzzy sets is evaluated, which is then defuzzified to get the acceleration command. A fuzzy-based guidance law for missiles has also been proposed by Creaser *et al.* (1998) using an evolutionary computing-based approach. The proposed law uses a genetic algorithm to generate a set of rules for the missile guidance law. Menon *et al.* (1998) uses fuzzy logic weightings to blend three well-known guidance laws to obtain enhanced homing performance. The composite law evaluates the weights on each of the guidance laws to obtain a blended guidance command for the missile. Yang and Chen (2001) have implemented an Hoo based guidance law. Unlike other guidance laws, it does not require the information of target acceleration, while ensuring acceptable interceptive performance for arbitrary target with finite acceleration.

3.1. *LOS Guidance.* LOS is the most widely used guidance strategy today. In fact, almost all guidance laws in use today have some form of LOS guidance because of its simplicity and ease of implementation. The LOS guidance employs the line of sight angle λ between the vehicle and the target which can easily be evaluated using Equation 1.

$$\lambda = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right), \tag{1}$$

where: (x_1, y_1), (x_2, y_2) are the missile and target position coordinates respectively.

The objective of the guidance system is to constrain the missile to lie as nearly as possible on the LOS. Since the missile ideally always lies on the line joining it to the

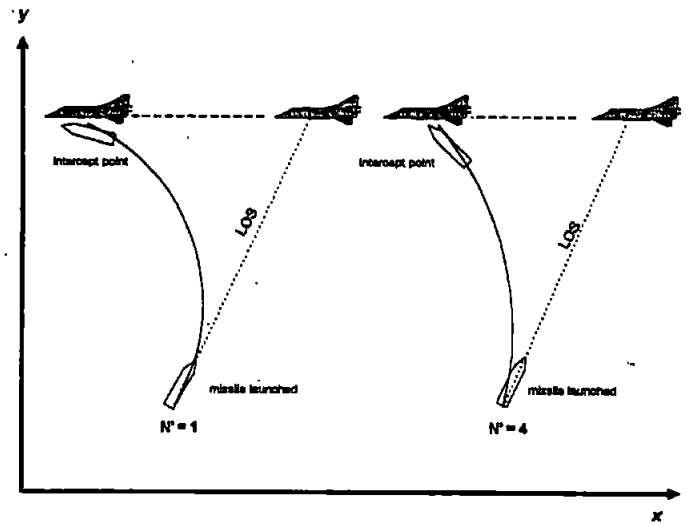


Figure 2. Proportional Navigation Guidance for a missile system.

target, the flight path will be a curved one. LOS guidance does not work well with manoeuvring targets. Also, the interception time is high, which can be abridged using different strategies as discussed in the following sections.

3.2. *Proportional Navigation Guidance (PNG) and its variants.* The Lark missile that was tested in 1950 was the first missile to use PNG. Since then, the PNG law has been used in virtually all of the world's tactical radar, infra-red and TV-guided missiles (Rajasekhar and Sreenatha, 2000). It is the most common and effective technique in case of non-manoeuvring targets that seeks to nullify the angular velocity of the LOS angle. The missile heading rate is made proportional to the LOS rate from the missile to the target. The rotation of the LOS is measured by a sensor (either onboard or from a ground station), which causes commands to be generated to turn the missile in the direction of the target. Mathematically the PNG law can be stated as:

$$\eta_c = N' V_c \dot{\lambda}, \tag{2}$$

where: η_c is the acceleration command, N' is the navigation ratio, V_c is the closing velocity and $\dot{\lambda}$ is the LOS angle rate. The advantage of using PNG over LOS guidance is that the interception time can be greatly reduced by adjusting the navigation constant as shown in Figure 2 for the case of $N'=1$ and $N'=4$. In the latter case, the missile steering commands are four times as great. As a result the missile veers off much more to the left resulting in engagement.

PNG, like LOS guidance, does not work well in the case of manoeuvring targets. However, the interception time is reduced. Augmented PNG (APNG) is a modified form of PNG to deal with target manoeuvres. Other forms of PNG are velocity compensated PNG (VCPNG), pursuit plus PNG, and dynamic lead guidance (Lin, 1991).

3.3. *Optimal Guidance Law.* Recently, great interest has been shown in using optimal control theory in the missile guidance problem. Two important mission parameters, missile target engagement time and the energy needed, can be reduced by utilising optimal control. Tsao and Lin (2000) proposed an optimal guidance law for short-range homing missiles to intercept highly manoeuvrable targets. The guidance problem that needs to be solved for the interception is to find the optimal missile trajectory such that the total time for the interception is minimised. The performance index used in the proposed optimal law is:

$$J = t_f = \int_0^{t_f} dt, \tag{3}$$

where: t_f is the interception time.

The proposed guidance law achieves the best performance in terms of the miss distance and interception time in comparison to the true proportional navigation guidance (TPNG) and APNG. However, a major disadvantage of this law is that the target's future trajectory must be known in advance, which is impossible to evaluate in a realistic environment (Tsao and Lin, 2000). A comprehensive review of optimal guidance laws is presented in (Lin, 1991).

4. AUTONOMOUS UNDERWATER VEHICLES.

AUVs are no longer engineering curiosities. They have been under development for over three decades and, in the last few years, there have been significant advances towards their use in operational missions (Millard and Griffiths, 1998). Although remotely operated vehicles (ROVs) play an important role in the offshore industry, their operational effectiveness is limited by the tethered cable and the reliance and cost of some kind of support platform. Given these limitations, developments in advance control engineering theory and the computation hardware for analysis, design and implementation, interest in the viability of employing AUVs in operational missions has been revived. The use of AUVs is increasingly being considered for applications such as cable/pipeline tracking, mines clearing operations, deep sea exploration, feature tracking etc.

The potential usage of AUVs is restricted by two main factors. The first is the limitation of battery power, which limits the use of AUVs for long duration missions. Most current vehicles use car batteries that need to be recharged every few hours, and this makes them unsuitable for long duration missions. The second limiting factor is associated with the current generation of onboard NGC systems. The vehicle must have a reliable and well-integrated NGC system of which guidance is the key element.

4.1. *Guidance Laws for AUVs.* The classical autopilots for AUVs are designed by controlling the heading or course angle in the control loop. By including an additional loop in the control system with position feedback from the sensors, an AUV guidance system can be designed. The guidance system generates reference trajectories to be followed by the vehicle utilising the data gathered by the navigation system. The following section presents some important guidance laws found in the literature.

4.1.1. *Waypoint Guidance by LOS.* Waypoint Guidance is the most widely used scheme in the field of AUVs. In the key paper by Healey and Lienard (1993), guidance is achieved between two points $[x_d(t_o), y_d(t_o)]$ and $[x_d(t_f), y_d(t_f)]$ by splitting the path between them into a number of waypoints $[x_d(k), y_d(k)]$ for $k = 1, 2, \ldots, N$ as shown in

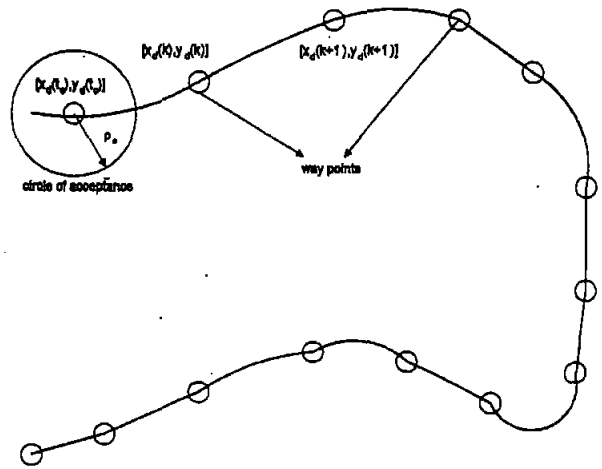


Figure 3. Way Point Guidance by LOS.

Figure 3. It is assumed that the vehicle is moving forward with speed U, then the LOS in terms of desired heading angle λ can be defined as:

$$\lambda = \tan^{-1}\left[\frac{y_d(k) - y(t)}{x_d(k) - x(t)}\right]. \tag{4}$$

where: $[x(t), y(t)]$ is the current location of the vehicle. Care must be exercised to ensure that the heading angle λ is in the proper quadrant. Determining when the vehicle reaches the vicinity of a waypoint is achieved by checking if the AUV lies within a circle of acceptance with radius ρ_o around the waypoint $[x_d(k), y_d(k)]$. If the vehicle's current location $[x(t), y(t)]$ satisfies:

$$\rho^2 = [x_d(k) - x(t)]^2 + [y_d(k) - y(t)]^2 \leq \rho_o^2, \tag{5}$$

the next waypoint $[x_d(k+1), y_d(k+1)]$ is selected. Typically, the circle of acceptance could be taken as two times the length of the vehicle (Healey and Lienard, 1998).

If on the other hand, $d\rho/dt$ goes from negative to positive without the above condition being met, then the waypoint has not been reached. At this point, the guidance law must decide whether to keep the same destination waypoint and direct the vehicle to the circle or choose the next depending on mission planning decisions. A major disadvantage of waypoint guidance is the undesirable consumption of control energy due to possible overshoots during the change of trajectory. So, selection of the reference trajectory for tracking is important to reduce the overshoot path width and thus to decrease the control energy consumption. Yeo *et al.* (1999) employ turning simulation to determine modified waypoints to avoid overshoot. Aguiar *et al.* (1998) and others (Aguiar and Pascoal, 1997) proposed a modification in waypoint guidance to deal with the presence of ocean currents. A current compensation for the heading autopilot has been developed which aligns the total vehicle velocity direction with the heading command.

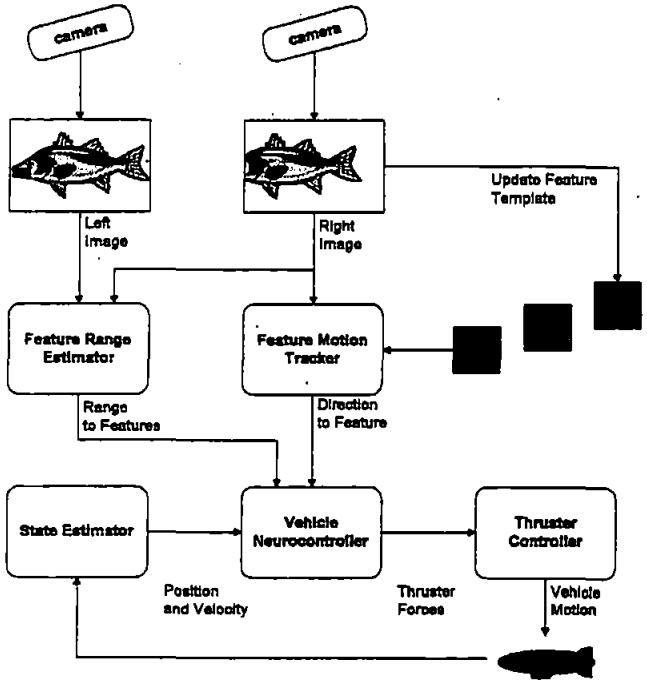


Figure 4. Vision-Based Guidance System for the Kambara AUV.

4.1.2. *Vision-Based Guidance.* The vision-based guidance technique has been inspired from the work of ROV operators, which utilise or rely on the visual information to perform tasks thus making a strong argument that visual imagery could be used to guide an AUV. Vision-based guidance has been mainly employed for cable tracking and docking problems (Gaskett *et al.*, 1999; Balasuriya and Ura, 1998; Briest *et al.*, 1997; Rock *et al.*, 1992). Briest *et al.* (1997) suggest an optical terminal guidance scheme for the docking of an AUV using a beacon. The beacon could be a light-emitting device, which can be identified using photo detectors onboard the AUV. This scheme is analogous to a heat-seeking air-to-air missile when locked on to its target. The disadvantage of using a beacon is that in shallow waters, especially during the daylight, the photo detectors can lock on to sunlight. The remedy could be to adjust the frequency of the light emitted by the beacon.

Gaskett *et al.* (1999) proposed vision-based guidance for an AUV named *Kambara* using two cameras. The authors demonstrated that guidance could be achieved by a feature tracker algorithm that requires two correlation operations within the feature tracker as shown in Figure 4. The feature motion tracker follows each feature between previous and current images from a single camera while the feature range estimator

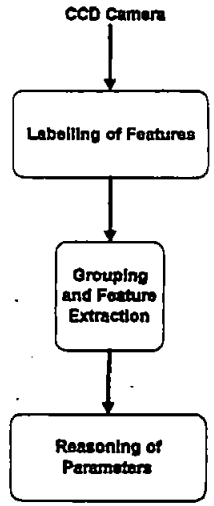


Figure 5. Vision-Based Guidance System for the Twin-Burger 2 AUV.

correlates between the left and right camera images. The feature motion tracker correlates stored feature templates to determine the image location and thus the direction to each feature. Range is determined by correlating the features in both images to find their pixel disparity. This pixel disparity is related to an absolute range using camera extrinsic and intrinsic parameters, which are determined by calibration. The direction and range to each feature is then fed to the controller, which determines a set of thruster commands. To guide the AUV, thruster commands become a function of the position of visual features.

A major drawback of using visual systems in underwater guidance is that the performance degrades in case of turbid water or when a cable is buried or there might be other similar cables appearing in the image. For such cases, a multi-sensor fusion technique has been proposed (Balasuriya and Ura, 1999a and b, 2000 and 2001). The proposed sensor fusion technique uses dead reckoning position uncertainty with a 2D-position model of the cable to predict the region of interest in the image captured by a camera mounted on an AUV. The 2D-position model of the layout of the cable is generated by taking the position co-ordinates (x_i, y_i) of a few points along the cable, which is then used to predict the most likely region of the cable in the image.

As opposed to the two camera approach, Balasuriya and Ura (1998) proposed a vision-based guidance law using a single camera. The technique has been implemented in a test-bed underwater robot, Twin-Burger 2, at the University of Tokyo for cable tracking and following a moving object. The basic idea underlying these schemes is that, the feature to be tracked introduces a particular geometric feature in the image captured by the CCD camera. The vision processor then labels these features, extract their location in the image and interprets the appearance into a guidance parameter as shown in Figure 5. For example, an underwater cable introduces a line feature in the

image, and the edges of a cylinder introduce a rectangle. The vision processor derives the equation of the line representing the cable in the image plane given by Equation 6, which gives the direction 'q' and position 'r' parameters.

$$r = x\cos(q) + y\sin(q). \tag{6}$$

where: (x, y) are the co-ordinates of the straight line equation.

In the case of a cylindrical object, the co-ordinates of the centroid of the object (rectangle) in the image plane and the area A covered by the object are derived. These parameters are then fused with other sensory parameters to determine the control references for the underwater vehicle.

Rock *et al.* (1992) devised a vision-based system to track a dot of light generated by a laser. The hardware comprises two cameras, one of which is used to locate the target. The vision system works by scanning the image from the last known location of the target, or from the centre of the screen if the target was not previously in view. The pixels are examined row by row, expanding outward towards the edge. If a target is found, its angle and elevation with respect to the centre of the image is evaluated and transmitted to the vision processor, while range can be found using successive images from both cameras. The proposed law has been proved to be valid only in the case of a single distinguishable target.

4.1.3. *Lyapunov-Based Guidance.* A Lyapunov function can be considered as a generalisation of the concept of distance or energy. The Lyapunov theorem states that, if the distance of the state along any trajectory of $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$ decreases with time, then $\mathbf{x}(t)$ must tend to 0 as $t \to \infty$ (Chen, 1984). Caccia *et al.* (2000) uses the concept to develop a new guidance law for unmanned underwater vehicles for testing on a prototype ROV, *Romeo*. This law is termed as a *medium range manoeuvring guidance law*. In this law, the vehicle is allowed to move from point (x, y) to (x_d, y_d) with a desired orientation Ψ_d as shown in Figure 6. By choosing the desired vehicle speeds

$$u_d = \zeta e\cos\alpha, \tag{7}$$

$$v_d = 0, \tag{8}$$

$$r_d = \mu\alpha + \zeta\frac{\cos\alpha\sin\alpha}{\alpha}(\alpha + h\theta), \tag{9}$$

(where ζ, μ and h are the tuning parameters) a Lyapunov function is suggested given by Equation 10, which makes the distance e between the two points converge to zero for increasing time.

$$V = \tfrac{1}{2}e^2 + \tfrac{1}{2}(\alpha^2 + h\theta^2), \tag{10}$$

where:

$$e = \sqrt{(x_d - x)^2 + (y_d - y)^2}, \tag{11}$$

$$\theta = \gamma - \psi_d, \quad \text{and} \tag{12}$$

$$\alpha = \gamma - \psi. \tag{13}$$

The parameters u_d, v_d and r_d are the desired vehicle's surge, sway and yaw velocities respectively. If an obstacle is detected along the way by the sensors with some

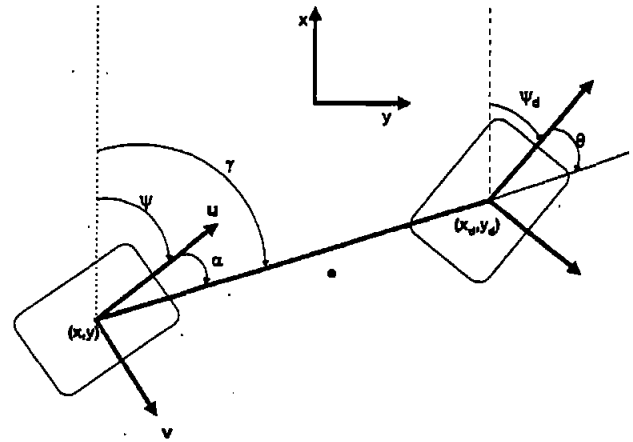


Figure 6. Medium Range Manoeuvring Guidance Law.

orientation δ and range d from the robot, the vehicle follows its profile until a suitable detaching condition is verified, and the vehicle then continues its free space manoeuvring. For feature following, the proposed law does not require control of the vehicle sway velocity while controlling the surge and yaw velocities.

4.1.4. *Guidance with Chemical Signals.* Using the fact that marine animals make extensive use of underwater chemical signals to avoid predators and to locate food sources etc., an interesting guidance scheme for AUVs using chemical signals has been proposed by Consi *et al.* (1994). The authors have built a small underwater robot, which mimics the chemical sensing abilities of a *lobster*. This class of robots is named as *biomimics* and are designed to mimic certain features of animals and act as animal substitutes in behavioral and neurobiological studies.

The goal of the research was to use the information in chemical signals to locate the source of a chemical discharge. In this respect, it has a number of scientific, environmental, commercial and defence related applications. The sensors used in the biomimic are conductivity sensors, and they are used to enable the AUV to follow a plume of saltwater in a freshwater flow-through flume. A simple gradient following algorithm is implemented to locate the source of discharge, which has the obvious disadvantage of getting trapped in local concentration minima and maxima.

4.1.5. *Proportional Navigation Guidance for AUVs.* Although PNG is widely used for missile guidance systems, Ahmad *et al.* (submitted) demonstrated that it could be tailored to work for AUVs as well. The authors propose a two-stage problem formulation to retrieve a returning AUV to the mother submarine. In the first stage, interception of the target (mother submarine) by the AUV is considered using a PNG law, which is the theme of the paper. In the second stage, the docking of the AUV is considered when in close proximity to the mother submarine and is an area of current investigation. The idea behind using PNG is that, if the AUV is made to lie on the LOS and hold there as well, a constant relative bearing between the AUV and target is ensured i.e., the LOS does not rotate and interception will occur. The PNG law

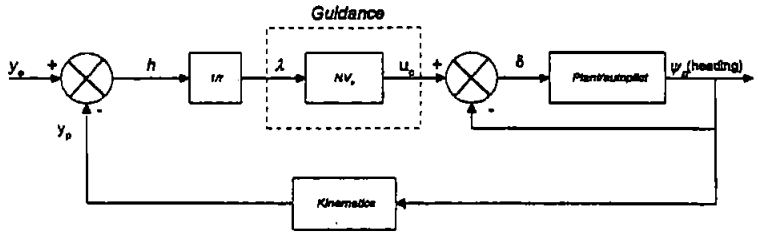


Figure 7. Proportional Navigation Guidance Loop.

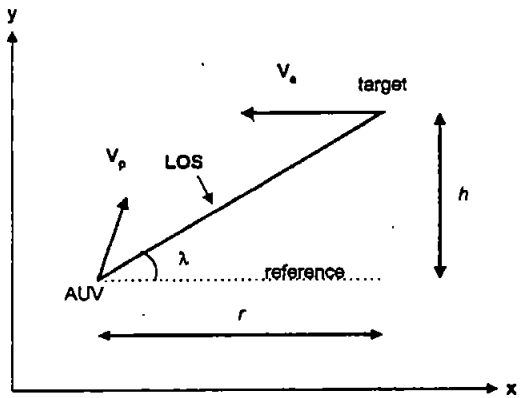


Figure 8. AUV-target engagement geometry.

can be stated as:

$$u_c \propto \lambda, \tag{14}$$

$$u_c = k\lambda, \tag{15}$$

where: k is the navigation constant, λ is the LOS angle and u_c is the command input.

The command input in this case is the heading angle ψ_c, therefore:

$$\psi_c = NV_c\lambda, \tag{16}$$

where: V_c is the closing velocity, and N is an important tuning parameter. The block diagram of the proposed guidance and control system is shown in Figure 7 and the AUV target engagement geometry is shown in Figure 8. The guidance system used is PNG, which generates commands for the control system. Different engagement scenarios have been considered. For stationary targets, the scheme is analogous to the waypoint guidance. For mobile targets, the PNG law generates suitable trajectories to be followed by the AUV for docking purposes.

4.1.6. *Guidance using Magnetometers for Cable Tracking.* The underwater cable network and its capacity are expanding very rapidly, and its installation and

maintenance is becoming more important. AUVs could be a potential tool for underwater cable tracking especially in case of deep waters where human intervention is not possible. Different schemes have been proposed for underwater cable tracking. Balasuriya and Ura (1998) proposed vision-based guidance for cable/pipeline tracking as outlined in Section 4.1.2., but in the case of shallow waters, where cables are buried to avoid being damaged by fishing gear or anchors, the performance degrades. For buried electrical or telecommunications cables, the remedy is to use on-board magnetometers, which can detect the magnetic field induced from the current flowing in the cable. The data from the magnetometer is fed to a cable locator that estimates the direction, burial depth and the distance of the vehicle from the cable. The data from the cable locator is then used to guide the vehicle. The *Aqua Explorer 1000* is an example of a successful implementation of magnetometer-based guidance for underwater cable tracking (Asakawa *et al.*, 1996; Kato *et al.*, 1994; Ito *et al.*, 1994). Guidance using magnetometers has limited applications as it can only be used to guide the vehicle towards the source of the magnetic field.

4.1.7. *Electromagnetic Guidance.* A major disadvantage of using optical or visual guidance systems is that the response is only good in nonturbid, clear environments, and it is limited over a wide range of background lighting and water turbidity conditions. Also, the AUV must lie within the field of light emitted by the beacon on the cable or dock and must be oriented in such a way that the optical sensors can detect the light. Feezor *et al.* (2001) employed an electromagnetic guidance (EM) technique during the homing/docking mode of an AUV. The EM guidance system uses a magnetic field generated by the coils on the dock, which is sensed by the coils in the AUV. The guidance system provides the AUV not only the bearing to the dock, but also the angle of the AUV relative to the field lines and thus the angle relative to the dock entrance. The accuracy of the proposed system is less than 20 cm but the range is limited to 25–30 m. The proposed system is quite robust under almost all oceanographic phenomena.

5. DISCUSSION.

The following discussion presents some modifications in the guidance design for AUVs. It can be stated that the waypoint guidance, or specifically, LOS guidance is likely to remain a key feature of all present guidance systems and systems that follow. By utilising LOS, several other guidance laws can be conceptualised. For example, waypoint guidance could be utilised for a cable following problem considering several waypoints on the cable by introducing beacons at different lengths and then follow the beacons on the cable using onboard sensors. Inexpensive photodetectors could be employed to detect the light, which can be tuned to operate over different frequencies. However, the major drawback as discussed in Section 4.1.7. is that the performance degrades in case of turbid water or when there is light emanating from other sources. This approach is analogous to the heat seeking air-to-air missile.

Another strategy to accomplish a cable tracking mission is to pose the guidance as a two-stage problem i.e. using waypoint guidance while on the surface and vision-based guidance or any other existing guidance scheme while submerged. In this manner, the vehicle would have precise position co-ordinates on the surface from the Global Positioning System (GPS) so that it can accurately reach the vicinity of the area of interest. Dead reckoning could be employed to estimate the position of the AUV under

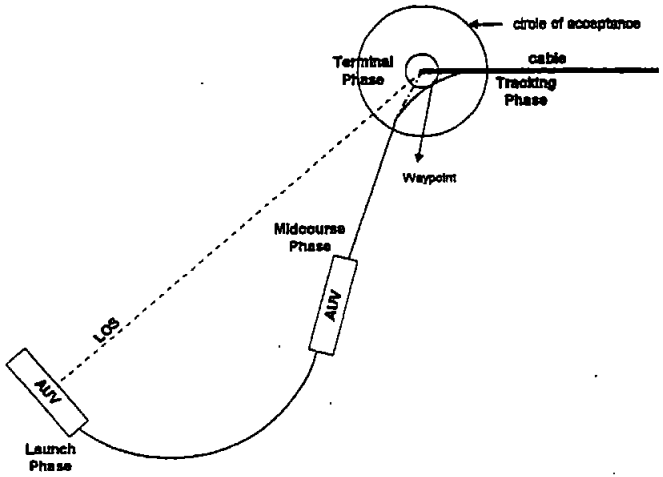


Figure 9. Planar View of the four phases of flight for cable tracking problem of an AUV.

water. A similar approach has been adopted by the Southampton Oceanography centre's AUTOSUB-1 (Mcphail and Pebody, 1998), which utilises GPS for position fixes on the surface and dead reckoning using an acoustic doppler current profiler, providing velocity measurement, while submerged.

A complete mission scenario for the underwater cable tracking problem could be to classify it into four different phases utilising different guidance laws. These are i) launch phase, ii) midcourse phase, iii) terminal phase, and iv) tracking phase as shown in Figure 9. In the first phase, called the launch phase or the boost phase, the vehicle is launched from a boat or from a mother submarine and guided in the direction of the LOS with maximum speed, using LOS guidance only. The heading command can be generated using Equation 4. Once the vehicle approaches the LOS, the midcourse phase could be invoked, in which the vehicle follows the LOS angle with maximum speed using waypoint guidance, Section 4.1.1. During this part of the operation, changes may be required to bring the vehicle onto the desired course and to make certain that it stays on that course. The midcourse guidance system is used to place the vehicle near the target area, where the system to be used in the final phase of guidance can take over. It should be noted that there is no need for the vehicle to submerge at this stage, as the objective is to approach the target area with maximum accuracy regardless of the orientation of the vehicle with respect to the cable. When the vehicle comes within the circle of acceptance, the third phase called the terminal phase is invoked. In this phase, the vehicle must be slowed down and submerged in order to line up with the cable/pipeline as shown in Figure 9. The circle of acceptance in this case, as opposed to that quoted in Healey and Lienard (1993) should be taken as at least the minimum turning radius of the vehicle in order to avoid overshoot. Finally the fourth phase, called the tracking phase, is launched utilising any existing guidance law. For example, the vehicle could use a vision-based guidance system to follow the cable outlined in Section 4.1.2. If the cable to be followed is an electrical/communication

cable, then magnetometers could be used to detect the radiation from the cable and guide the vehicle in the appropriate direction.

6. CONCLUDING REMARKS. This paper presents several guidance laws for autonomous vehicles with emphasis on AUVs. Guidance laws for airborne missile systems are also explored. It has been shown that the guidance system plays the vital role in bringing autonomy to the whole system. It is observed that most of the current AUV systems employ either the classical waypoint guidance to reach a target area or the more advanced vision-based guidance for cable/pipeline tracking. In practice, LOS guidance is the key element of all guidance systems. Some hybrid guidance schemes are also proposed based on existing airborne and underwater guidance laws.

REFERENCES

Aguiar, A. and Pascoal, A. (1997). Modelling and control of an autonomous underwater shuttle for the transport of benthic laboratories. *Oceans'97, MTS/IEEE Conference Proceedings, Halifax, Canada*, Vols 1 & 2, pp. 888–895, October.

Aguiar, A., Oliveira, P., Silvertre, C. and Pascoal, A. (1998). Guidance and control of the SIRENE underwater vehicle: from system design to tests at sea. In *Proc. Of Oceans'98, Nice, France*, Vols 1–3, pp. 1043–1048, September.

Ahmad, S. M., Sutton, R. and Burns, R. S. Retrieval of an autonomous underwater vehicle: an interception approach. *Submitted to the Journal of Underwater Technology*.

Asakawa, K., Kojima, J., Ito, Y. and Takagi, S. (1996). Autonomous underwater vehicle AQUA EXPLORER 1000 for inspection of underwater cables. *Proc. of the 1996 Symposium on Autonomous Underwater Vehicle Technology*, pp. 10–17, IEEE.

Balasuriya, A. and Ura, T. (1998). Autonomous target tracking by underwater robots based on vision. *Proceedings of IEEE UT'98*, pp. 191–197, Tokyo, Japan, April.

Balasuriya, A. and Ura, T. (1999a). Multisensor fusion for autonomous underwater cable tracking. *Oceans'99 MTS/IEEE*, pp. 209–215.

Balasuriya, A. and Ura, T. (1999b). Sensor fusion technique for cable following by autonomous underwater vehicles. *Proceedings of the 1999 IEEE International Conference on Control Application, Kohala Coast-Island of Hawai'I, USA*, pp. 1779–1784.

Balasuriya, A. and Ura, T. (2000). Autonomous target tracking by Twin-Burger 2. *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 849–854.

Balasuriya, A. and Ura, T. (2001). Underwater cable following by Twin-Burger 2. *Proceedings of the 2001 IEEE International Conference on Robotics and Automation, Seoul, Korea*, pp. 920–925.

Briest, S., Cowen, S. and Dombrowski, J. (1997). Underwater docking of autonomous undersea vehicles using optical terminal guidance. *Oceans'97 MTS/IEEE Conference Proceedings, Halifax, Canada*, Vols 1 & 2, pp. 1143–1147, October.

Bryson, A. E. Jr. and Ho, Y. C. (1969). *Applied Optimal Control*. Blaisdell, Waltham, MA, USA.

Caccia, M., Bruzzone, G. and Veruggio, G. (2000). Guidance of unmanned underwater vehicles: experimental results. *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, pp. 1799–1904. IEEE, April.

Chen, C. T. (1984). Linear System Theory and Design. *CBS College Publishing*.

Cloutier, J. R., Evers, J. H. and Feeley, J. J. (1989). Assessment of air-to-air missile guidance and control technology. *IEEE Control Systems Magazine*, 9(6), pp. 27–34, October.

Consi, T. R., Goudey, C. A., Cho, J., Atema, J. and Chryssostomidis, C. (1994). AUV guidance with chemical signals. *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology*, pp. 450–455, July.

Creaser, P. A., Stacey, B. A. and White, B. A. (1998). Evolutionary generation of fuzzy guidance laws. *UKACC International Conference on Control'98, UK*, Vol. II, no. 455: pp. 883–888, September 1–4.

Feezor, M. D., Sorrell, F. Y., Blankinship, P. R. and Bellingham, J. G. (2001). Autonomous underwater vehicle homing/docking via electromagnetic guidance. *IEEE Journal of Oceanic Engineering*, 26(4), pp. 515–521, October.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley & Sons.

Gaskett, C., Wettergreen, D. and Zelinsky, A. (1999). Autonomous guidance and control for an underwater robotic vehicle. *International Conference on Field and Service Robotics, FSR'99, Pittsburgh, USA*, 29–31 August.

Healey, A. J. and Lienard, D. (1993). Multivariable sliding model control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering*, 18(3), pp. 327–339, July.

Ito, Y., Kato, N., Kojima, J. and Takagi, S. (1994). Cable tracking for autonomous underwater vehicle. *Proc. of the 1994 Symposium on Autonomous Underwater Vehicle Technology*, pp. 218–224, IEEE, 1994.

Kato, N., Ito, Y., Kojima, J. and Takagi, S. (1994). Control performance of autonomous underwater vehicle 'AQUA EXPLORER 1000' for inspection of underwater cables. *Proc. Oceans Engineering for Today's Technology and Tomorrow's Presentation*, vol. 1, pp. I/135–I/140, IEEE.

Lin, C. F. (1991). *Modern Navigation, Guidance and Control Processing, Volume II*. Prentice Hall.

Lin, C. L. and Su, H. W. (2000). Intelligent control theory in guidance and control system design: an overview. *Proc. Natl. Sci. Counc. ROC(A). (Invited Review Paper)*, 24(1), pp. 15–30.

Locke, A. S. (1955). Guidance. D. Van Nostrand Co., Princeton, NJ, USA.

Mcphail, S. D. and Pebody, M. (1998). Navigation and control of an autonomous underwater vehicle using a distributed, networked control architecture. *Journal of the Society for Underwater Technology*, 23(1), pp. 19–30.

Menon, P. K. and Iragavarapu, V. R. (1998). Blended homing guidance law using fuzzy logic. *AIAA Guidance, Navigation and Control Conference, Boston MA*, August 10–12.

Millard, N. W. and Griffiths, G. (1998). Versatile autonomous submersibles – the realising and testing of a practical vehicle. *Journal of the Society for Underwater Technology*, 23(1), pp. 7–17.

Nazaroff, G. J. (1976). An optimal terminal guidance law. *IEEE Trans. Automat. Contr.*, 21(6), pp. 407–408.

Potter, J. E. (1964). *A guidance-navigation separation theorem*. AIAA Paper 64-653, AIAA, Washington DC, USA.

Rajasekhar, V. and Sreenatha, A. G. (2000). Fuzzy logic implementation of proportional navigation guidance. *Acta Astronautica, Elsevier Science Ltd.*, 46(1), pp. 17–24.

Rock, S. M., Lee, M. J., Wang, H. H., Marks, R. L. and Burton, R. C. (1992). Combined camera and vehicle tracking of underwater objects. In *Proceedings of ROV'92, San Diego, CA, USA*, pp. 1–8, June.

Speyer, J. L., Greenwell, W. M. and Hull, D. G. (1982). Adaptive noise estimation and guidance for homing missile. *AIAA Guid. and Contr. Conf.*, Washington DC, USA.

Sutton, R., Burns, R. S. and Craven, P. J. (2000). Intelligent steering control of an autonomous underwater vehicle. This *Journal*, 53, 511–525.

Tsao, L. P. and Lin, C. S. (2000). A new optimal guidance law for short-range homing missiles. *Proceedings of the National Science Council, Republic of China*, 24(6), pp. 422–426.

Yang, C. D. and Chen, H. Y. (2001). Three-dimensional nonlinear H-oo guidance law. *International Journal of Robust and Nonlinear Control*, 11(2), pp. 109–129.

Yeo, D. J. (1999). Design of AUV Tracking System using the Sliding Mode Control and the Optimal Control Theory. *MS Thesis, Department of Naval Architecture and Ocean Engineering, Seoul National University, Korea.*

Zarchan, P. (1994). *Tactical and Strategic Missile Guidance*, 2nd Ed. AIAA, Inc., Washington DC, USA.

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Control Engineering Practice 12 (2004) 1531–1539
CONTROL ENGINEERING PRACTICE
www.elsevier.com/locate/conengprac

Adaptive tuning of a Kalman filter via fuzzy logic for an intelligent AUV navigation system

D. Loebis*, R. Sutton, J. Chudley, W. Naeem

Marine and Industrial Dynamic Analysis Research Group, School of Engineering, The University of Plymouth, Drake Circus, Plymouth PL4 8AA, UK

Received 17 August 2003; accepted 27 November 2003

Abstract

This paper describes the implementation of an intelligent navigation system, based on the integrated use of the global positioning system (GPS) and several inertial navigation system (INS) sensors, for autonomous underwater vehicle (AUV) applications. A simple Kalman filter (SKF) and an extended Kalman filter (EKF) are proposed to be used subsequently to fuse the data from the INS sensors and to integrate them with the GPS data. The paper highlights the use of fuzzy logic techniques to the adaptation of the initial statistical assumption of both the SKF and EKF caused by possible changes in sensor noise characteristics. This adaptive mechanism is considered to be necessary as the SKF and EKF can only maintain their stability and performance when the algorithms contain the true sensor noise characteristics. In addition, fault detection and signal recovery algorithms during the fusion process to enhance the reliability of the navigation systems are also discussed herein. The proposed algorithms are implemented to real experimental data obtained from a series of AUV trials conducted by running the low-cost *Hammerhead* AUV, developed by the University of Plymouth and Cranfield University.

Keywords: Autonomous underwater vehicles; Navigation; Sensor fusion; Kalman filters; Extended Kalman filters; Fuzzy logic

1. Introduction

The development of autonomous underwater vehicles (AUVs) for scientific, military and commercial purposes in applications such as ocean surveying (Størkersen, Kristensen, Indreeide, Seim, & Glancy, 1998), unexploded ordnance hunting (Wright et al., 1996) and cable tracking and inspection (Asakawa, Kojima, Kato, Matsumoto, & Kato, 2000) requires the corresponding development of navigation systems. Such systems are necessary to provide knowledge of vehicle position and attitude. The need for accuracy in such systems is paramount: erroneous position and attitude data can lead to a meaningless interpretation of the collected data or even to a catastrophic failure of an AUV.

A growing number of research groups around the world are developing integrated navigation systems utilising inertial navigation system (INS) and global positioning system (GPS) (Gade & Jalving, 1999; Grenon, An, Smith, & Healey, 2001; Yun et al., 1999). However, few of these works make explicit the essential need for fusion of several INS sensors that enable the users to maintain the accuracy or even to prevent a complete failure of this part of the navigation system, before being integrated with the GPS. Kinsey and Whitcomb (2003), e.g. use a switching mechanism to prevent a complete failure of the INS. Although simple to implement, the approach may not be appropriate to use to maintain a certain level of accuracy.

Several estimation methods have been used in the past for multisensor data fusion and integration purpose (Loebis, Sutton, & Chudley, 2002). To this end, simple/extended Kalman filter (SKF/EKF) and their variants have been popular methods in the past and interest in developing the algorithms has continued to the present day. However, a significant difficulty in designing a SKF/EKF can often be traced to incomplete a priori knowledge of the process covariance matrix (Q) and measurement noise covariance matrix (R). In most practical applications, these matrices are initially estimated or even unknown. The problem here is that the optimality of the estimation algorithm in the SKF/EKF

*Corresponding author. Tel.: +44-1752232633; fax: +44-1752232638.
E-mail address: d.loebis@plymouth.ac.uk (D. Loebis).

0967-0661/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2003.11.008

setting is closely connected to the quality of a priori information about the process and measurement noise (Mehra, 1970, 1971). It has been shown that insufficiently known a priori filter statistics can reduce the precision of the estimated filter states or introduces biases to their estimates. In addition, incorrect a priori information can lead to practical divergence of the filter (Fitzgerald, 1971). From the aforementioned, it may be argued that the conventional SKF/EKF with fixed R and/or Q should be replaced by an adaptive estimation formulation as discussed in the next section.

2. The adaptive tuning of Kalman filter algorithm

In the past few years, only a few publications in the area of adaptive Kalman filtering can be found in the literature. The two major approaches that have been proposed for adaptive Kalman filtering are multiple model adaptive estimation (MMAE) and innovation adaptive estimation (IAE). Although the implementation of these approaches are quite different, they both share the same concept of utilising new statistical information obtained from the innovation (or residual) sequence. In both cases, the innovation Inn_k at sample time k is the difference between the real measurement z_k, received by the filter and its estimated (predicted) value $\hat{z}_k$. The predicted measurement is the projection of the filter predicted states $\hat{x}_k^-$ onto the measurement space through the measurement design matrix H_k. Innovation represents additional information available to the filter as a result of the new measurement z_k. The occurrence of data with statistics different from the a priori information will first show up in the innovation vector. For this reason, the innovation sequence represents the information content in the new observation and is considered the most relevant source of information to the filter adaptation. Interested readers can refer to (Kailath, 1968a, b, 1970) for a more detailed discussion of the innovation sequence and its use in linear filter theory.

In the MMAE approach, a bank of Kalman filters runs in parallel (Magill, 1965; Hanlon & Maybeck, 2000) or with a gating algorithm (Chaer, Bishop, & Ghough, 1997) under a different model for the statistical filter information matrices, i.e. Q and R. In the IAE approach (Mehra, 1970, 1971), the Q and R matrices themselves are adapted as measurements evolve with time. In this paper, the IAE approach coupled with fuzzy logic techniques with membership functions designed using heuristic methods is used to adjust the R matrix of both the SKF and EKF. The proposed algorithms in this paper are implemented using a set of experimental data obtained from the *Hammerhead* AUV trials conducted in July 2003 at Roadford Reservoir, Devon, UK. Initial work using purely simulated data on the proposed algorithms can be found in Loebis, Sutton, and Chudley, 2003b and Loebis, Chudley, and Sutton, 2003a.

2.1. Fuzzy simple Kalman filter

In this section, an on-line innovation-based adaptive scheme of the SKF to adjust the R matrix employing the principles of fuzzy logic is presented. The fuzzy logic is chosen mainly because of its simplicity. This motivates the interest in the topic, as testified by related papers which have been appearing in the literature (Kobayashi, Cheok, Watanabe, & Muneka, 1998; Jetto, Longhi, & Vitali, 1999; Loebis et al., 2003a, b). The fuzzy logic simple Kalman filter (FSKF) proposed herein and fuzzy logic extended Kalman filter (FEKF) discussed in Section 4 are based on the IAE approach using a technique known as covariance matching (Mehra, 1970). The basic idea behind the technique is to make the actual value of the covariance of the innovation sequences match its theoretical value.

The actual covariance is defined as an approximation of the Inn_k sample covariance through averaging inside a moving estimation window of size M (Mohamed & Schwarz, 1999) which takes the following form:

$$\hat{C}_{Inn_k} = \frac{1}{M}\sum_{j=j_0}^{k} Inn_k\, Inn_k^T, \tag{1}$$

where $j_0 = k - M + 1$ is the first sample inside the estimation window. An empirical experiment is conducted to choose the window size M. From experimentation, it was found that a good size for the moving window in (1) is 15.

The theoretical covariance of the innovation sequence is defined as

$$S_k = H_k P_k^- H_k^T + R_k. \tag{2}$$

The logic of the adaptation algorithm using covariance matching technique can be qualitatively described as follows. If the actual covariance value $\hat{C}_{Inn_k}$ is observed, whose value is within the range predicted by theory S_k and the difference is very near to zero, this indicates that both covariances match almost perfectly and only a small change is needed to be made on the value of R. If the actual covariance is greater than its theoretical value, the value of R should be decreased. On the contrary, if $\hat{C}_{Inn_k}$ is less than S_k, the value of R should be increased. This adjustment mechanism lends itself very well to being dealt with using a fuzzy-logic approach based on rules of the kind:

$$\text{IF} < \textit{antecedent} > \text{THEN} < \textit{consequent} >, \tag{3}$$

where antecedent and consequent are of the form $\chi \in O_i$, $\kappa \in L_i$, $i = 1, 2, \ldots$, respectively, where χ and κ are the input and output variables, respectively, and O_i and L_i are the fuzzy sets.

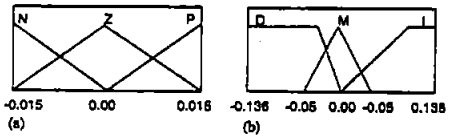


Fig. 1. Membership function of: (a) $delta_k$, and (b) ΔR_k.

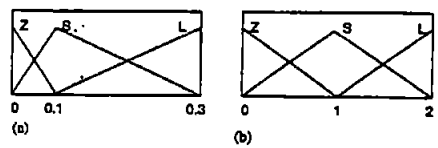


Fig. 2. Membership function of: (a) $|delta_k|$, and (b) R_k.

To implement the above covariance matching technique using the fuzzy logic approach, a new variable called $delta_k$, is defined to detect the discrepancy between $\hat{C}_{Inn_k}$ and S_k. It is important to note that in this particular application, $\hat{C}_{Inn_k}$ and S_k are constrained to be diagonal matrices. The following three fuzzy rules of the kind (3) are used:

$$\text{IF} < delta_k \cong 0 > \text{THEN} < R_k \text{ is unchanged} >, \tag{4}$$

$$\text{IF} < delta_k > 0 > \text{THEN} < R_k \text{ is decreased} >, \tag{5}$$

$$\text{IF} < delta_k < 0 > \text{THEN} < R_k \text{ is increased} >. \tag{6}$$

Thus R is adjusted according to

$$R_k = R_{k-1} + \Delta R_k, \tag{7}$$

where ΔR_k is added or subtracted from R at each instant of time. Here $delta_k$ is the input to the fuzzy inference system (FIS) and ΔR_k is the output.

On the basis of the above adaptation hypothesis, the FIS can be implemented using three fuzzy sets for $delta_k$; N = *Negative*, Z = *Zero* and P = *Positive*. For ΔR_k the fuzzy sets are specified as I = *Increase*, M = *Maintain* and D = *Decrease*. The membership functions of these fuzzy sets which are designed using a heuristic approach are shown in Fig. 1.

2.2. Sensor fault diagnostic and recovery algorithm

In addition to the adaptation procedure, the FSKF has been equipped with the sensor fault diagnostic and recovery algorithm as proposed by Escamilla-Ambrosio & Mort (2001). The basic idea behind this algorithm is that the amplitude of the actual value of the Inn_k and its theoretical value ($\sqrt{S_k}$) for a sensor without any fault must be around 1, but it increases abruptly if a transient or persistent fault is present in the measurement data. For this purpose a variable $InnC_k$ is defined as

$$InnC_k = \frac{|Inn_k|}{\sqrt{S_k}}. \tag{8}$$

Thus, if the value of $InnC_k$ is greater or equal than a threshold (α) then a transient fault is declared and Inn_k is assigned a value of 0. If $InnC_k$ is still greater than α for an instant of time, the persistent fault is declared and Inn_k is assigned a value of $\sqrt{S_k}$ multiplied by a random number. From experimentation, it was found that the good value of α is 1.2.

Table 1
Fuzzy rule base FLO

$\lvert delta_k \rvert$	R_k		
	Z	S	L
Z	G	G	AV
S	G	AV	P
L	AV	P	P

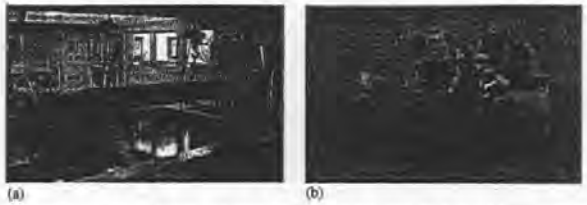
2.3. Fuzzy logic observer

To monitor the performance of a FSKF, another FIS called the fuzzy logic observer (FLO) (Escamilla-Ambrosio & Mort, 2001) is used. The FLO assigns a weight or degree of confidence denoted as c_k, a number on the interval [0,1], to the FSKF state estimate. The FLO is implemented using two inputs: the values of $|delta_k|$ and R_k. The membership functions of these variables were found using a heuristic method that produced a non-symmetrical shape for $|delta_k|$ and a symmetrical shape for R_k are shown in Fig. 2.

The fuzzy labels for the membership functions: Z = Zero, S = Small and L = Large. Three fuzzy singletons are defined for the output c_k and are labelled as G = Good, AV = Average and P = Poor with values 1, 0.5 and 0, respectively. The basic heuristic hypothesis for the FLO is as follows: if the value of $|delta_k|$ is near to zero and the value of R_k is near to zero, then the FSKF works almost perfectly and the state estimate of the FSKF is assigned a weight near 1. On the contrary, if one or both of these values increases far from zero, it means that the FSKF performance is degrading and the FLO assigns a weight near 0. Table 1 gives the complete fuzzy rule base of each FLO.

3. Fusion of INS sensor data

In this section, the FSKF algorithm is applied to the linear heading model of the *Hammerhead* AUV. Fig. 3(a) shows the vehicle before leak testing and



(a) (b)

Fig. 3. The *Hammerhead* AUV: (a) before leak testing and ballasting, and (b) during a system identification trial.

ballasting in the laboratory test tank and Fig. 3(b) shows the vehicle during a heading system identification trial. Inputs to the rudder of the vehicle are sent by the user from a laptop through an umbilical cable. Thus in reality it was travelling in a semi-autonomous model for that specific trial. The drag effect of the cable is considered to be negligible. An electronic compass and an inertial measurement unit (IMU) on board the vehicle are used to capture the corresponding responses.

The system matrix (A), input matrix (B), and output vector (H) of the linear discrete state space model (see Appendix A) obtained from the trial data are, respectively, $A = [0\ 1;\ -0.98312\ 1.9831]$, $B = [-0.0031961\ \ -0.0036115]$, $H = [1\ \ 0]$, with the yaw and delayed-yaw as the component of the states.

This model is assumed to be sufficiently accurate to represent the dynamics of the vehicle, and for this reason, any output produced by the model after being excited by an input, can be considered as an actual output value. This assumption also motivates the use of the model output as a reference in measuring the performance of the FSKF algorithms.

To test the FSKF algorithms, real data obtained from the electronic compass and IMU (Fig. 5), as a response to the input shown in Fig. 4, are fused together with two sets of simulated data. To produce the simulated data, the noise in Figs. 6(a) and (b) are simply added to the electronic compass and IMU real data, respectively. A possible real-time scenario that can result in the noise with the characteristic shown in Fig. 6(a) is that the second electronic compass is located in close proximity to the propeller DC motor whose internal temperature increases with time and affects the sensor ambient temperature. A similar scenario can also be considered to occur when the second IMU is located in close proximity to the front hydroplane stepper motor whose initial internal temperature is high and settles down after sometime. This particular scenario can result in the noise characteristic as shown in Fig. 6(b). The initial condition of the states are $[0\ \ 0]^T$, $P_0 = 0.01I_2$ (see Appendix A) and $Q = \mathrm{diag}(0, 0.1725 \times 10^{-7})$. The actual value of R for each sensor is assumed unknown, but its initial value is selected as 1. Simulation results are shown in the next section.

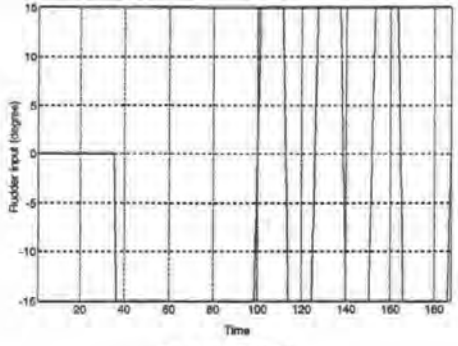


Fig. 4. Input rudder.

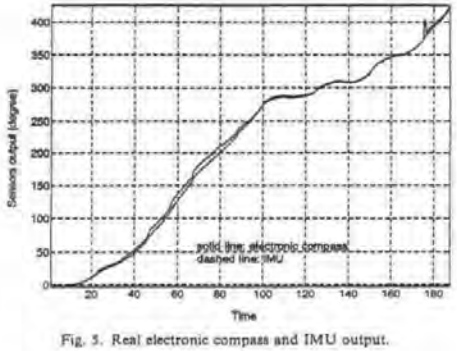


Fig. 5. Real electronic compass and IMU output.

3.1. Simulation result

Figs. 7 and 8 are the simulation results showing the response of the *Hammerhead* AUV observed by electronic compass and IMU, respectively, while Figs. 9 and 10 by sensor 3 and 4, which are the output of the two former sensors added with uniform noise increasing and

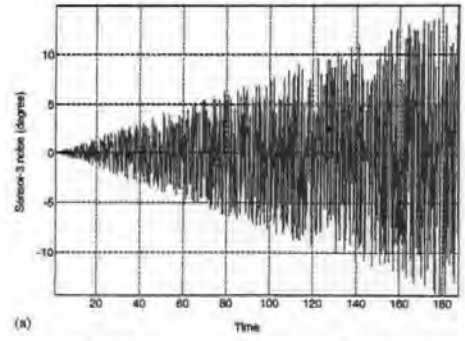


Fig. 6. (a) Added noise profile to the electronic compass data, and (b) to the IMU data.

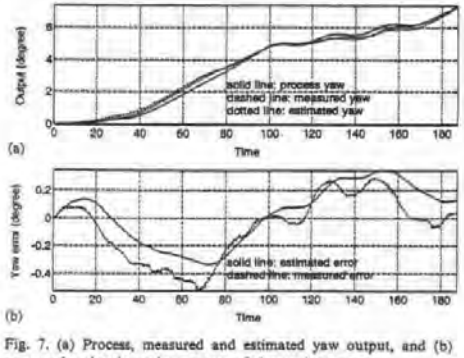


Fig. 7. (a) Process, measured and estimated yaw output, and (b) measured and estimated yaw error of electronic compass.

decreasing with time, respectively. Figs. 7 and 8 show improvements in the level of error produced by the proposed FSKF algorithms as compared to direct

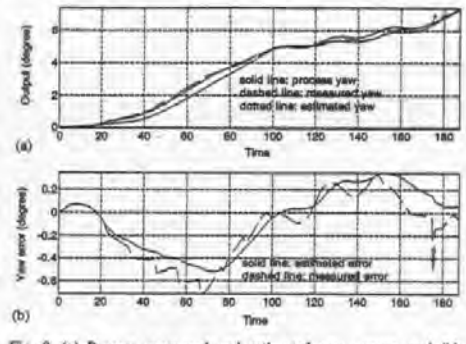


Fig. 8. (a) Process, measured and estimated yaw output, and (b) measured and estimated yaw error of IMU.

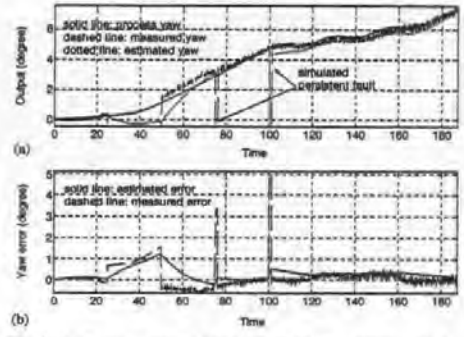


Fig. 9. (a) Process, measured and estimated yaw output, and (b) measured and estimated yaw error of sensor 3 (electronic compass + simulated noise).

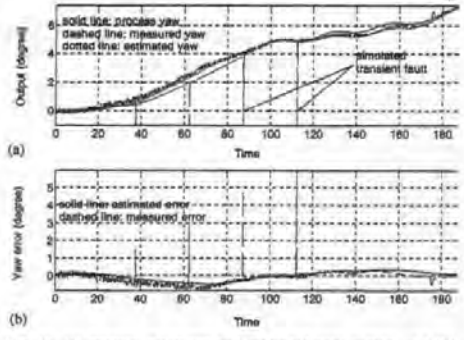


Fig. 10. (a) Process, measured and estimated yaw output, and (b) measured and estimated yaw error of sensor 4 (IMU + simulated noise).

measurements from the sensors. Apart from the improvements in the level of error, Figs. 9 and 10 also show how the proposed algorithms have detected transient and persistent faults in the sensors (see Section 2.2) and made an appropriate recovery.

To fuse the estimated yaw, a centre of gravity method is used

$$\hat{z}_k = \frac{\sum_{i=1}^{n} \hat{z}_{k_i} c_{k_i}}{\sum_{i=1}^{n} c_{k_i}}, \tag{9}$$

where $\hat{z}_{k_i}$ is the output of the ith FSKF ($i = 1, 2, 3, 4$) and c_{k_i} is the respective weight at instant time k. Fig. 11 shows the comparison of the actual and the fused estimated yaw. It is clear, by comparing Fig. 11 and Figs. 7–10 that an improvement has been achieved by fusing the estimated yaw.

Finally, the following performance measure are adopted for comparison purposes:

$$J_{zv} = \sqrt{\frac{1}{n}\sum_{k=1}^{n} (za_k - z_k)^2}, \tag{10}$$

$$J_{ze} = \sqrt{\frac{1}{n}\sum_{k=1}^{n} (za_k - \hat{z}_k)^2}, \tag{11}$$

where za_k is the actual value of the yaw, z_k is the measured yaw, $\hat{z}_k$ is the estimated yaw at an instant of time k and n the number of samples (Table 2).

A close look on the J_{zv} and J_{ze} on Table 2 of each sensor indicates that the FSKF has improved the accuracy of the heading information. The result of fusing the estimated data has shown a further improvement. A slight offset shown by the final fusion result might be caused by an inaccurate model of the process noise (see Appendix A) and its covariance (Q). Adaptation of these parameters is the topic of a future investigation. It should also be noted that from a theoretical point of view, the analysis of the stability of

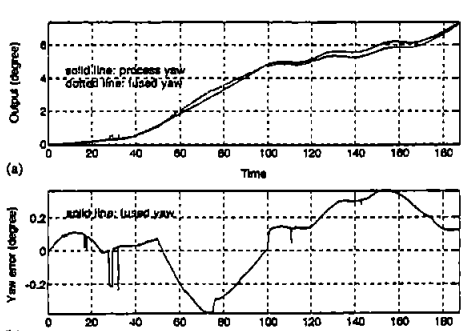


Fig. 11. (a) Process and estimated fused yaw output, and (b) fused yaw error.

Table 2
Comparison of performance

Sensor	Performance	
	J_{zv} (deg)	J_{ze} (deg)
Electronic compass	13.4050	12.1170
IMU	17.3507	15.8216
Electronic compass + noise	37.5725	23.6664
IMU + noise	22.0702	14.6159
Sensor fusion		11.9650

the FSKF needs to be investigated. However, this is not easily undertaken due to the use of the adaptation techniques used herein. Future work will address this issue more rigorously.

4. Integrated GPS/INS

Here, the fused estimated yaw obtained previously is treated as a single imaginary yaw sensor and used by other INS sensors to transform data from body co-ordinate to Earth co-ordinate frame where integration with GPS data is performed using a combination of fuzzy logic and EKF techniques and can be referred to as FEKF.

A continuous time model of the vehicle motion appropriate to this problem is taken to be

$$\dot{X}(t) = F(X(t)) + W(t), \tag{12}$$

$$Z(t) = H(X(t)) + V(t). \tag{13}$$

Denoted by $X(t) = [\lambda(t)\ \varphi(t)\ \psi(t)\ \vartheta(t)\ \varsigma(t)\ \upsilon(t)]^T$ is the model states. $\lambda(t)$ and $\varphi(t)$ are the longitude and latitude of the AUV position in Earth co-ordinate frame which are obtained from a GPS receiver, $\psi(t)$ is the yaw angle obtained from the imaginary yaw sensor, $\vartheta(t)$ is yaw rate, $\varsigma(t)$ and $\upsilon(t)$ are the surge and sway velocity, respectively.

In this system model, F and H are both continuous functions, continuously differentiable in $X(t)$. The $W(t)$ and $V(t)$ are both zero-mean white noise for the system and measurement models, respectively.

The model states are related through the following kinematically based set of functions ($F(X(t))$ in Eq. (12)):

$$\dot{\varsigma}(t) = 0, \tag{14}$$

$$\dot{\upsilon}(t) = 0, \tag{15}$$

$$\dot{\psi}(t) = \vartheta(t), \tag{16}$$

$$\dot{\vartheta}(t) = 0, \tag{17}$$

$$\dot{\lambda}(t) = \varsigma(t)\cos\psi(t) - \upsilon(t)\sin\psi(t), \tag{18}$$

$$\dot{\varphi}(t) = \varsigma(t)\sin\psi(t) + \upsilon(t)\cos\psi(t). \tag{19}$$

The output measurements are related through the states by the following output matrix:

$$H(X(t)) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{20}$$

when GPS signal is available, and when it is not,

$$H(X(t)) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{21}$$

To obtain an EKF with an effective state prediction equation in a simple form, the continuous time model of (14)–(21) has been linearised about the current state estimates, producing

$$\Phi(t) = \begin{bmatrix} 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & & 0 & 1 & 0 & 0 \\ 0 & 0 & & 0 & 0 & 0 & 0 \\ 0 & 0 & -\varsigma(t)\sin\psi(t) - \upsilon(t)\cos\psi(t) & 0 & \cos\psi(t) & -\sin\psi(t) \\ 0 & 0 & \varsigma(t)\cos\psi(t) - \upsilon(t)\sin\psi(t) & 0 & \sin\psi(t) & \cos\psi(t) \end{bmatrix} \tag{22}$$

and Γ is a matrix identical as in either (20) or (21). Subsequent discretisation with period $T = 0.125$ s of the linearised model results in an EKF algorithm similar to the SKF algorithms in Appendix A (where Φ and Γ are equivalent to A and H), only this time the Φ matrix is updated at every iteration. The initial conditions are $P_0 = 0.01\, I_6$ and Q is made constant as diag(10, 10, 1, 0.1, 0.1, 0.1). The actual value of R is assumed unknown but its initial value is selected as diag(1000, 1000, 5, 1, 2, 2)

The FSKF algorithm from Section 2 is then implemented, only this time the adaptation of the (i, i)th element of R_k is made in accordance with the (i, i)th element of $delta_k$. Here a single-input–single-output FIS as shown in Fig. 1, is used sequentially to generate the correction factors for the elements in the main diagonal of R_k as the following:

$$R_k(i, i) = R_{k-1}(i, i) + \Delta R_k. \tag{23}$$

Fig. 12 shows the *Hammerhead* AUV trajectory obtained using GPS, dead reckoning using INS sensors (through double integration of the accelerometer data with respect to time) and integrated GPS/INS. As the initial value of R for both $\lambda(t)$ and $\varphi(t)$ is 1000, the standard EKF algorithm puts less weight on the position obtained by GPS and more on the prediction

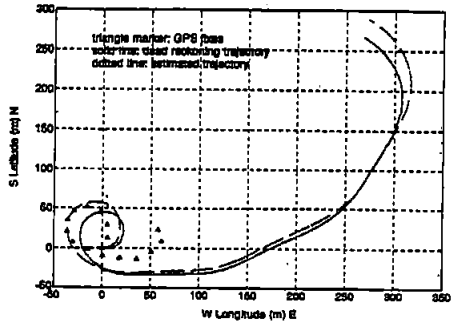


Fig. 12. AUV trajectory obtained using GPS, INS sensors (dead reckoning method) and GPS/INS using EKF without adaptation.

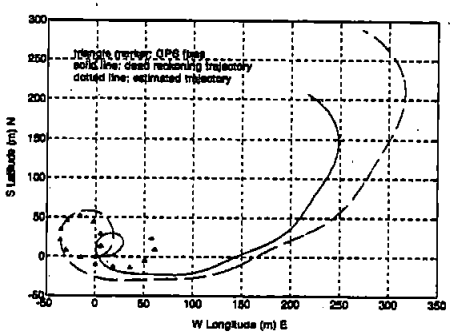


Fig. 13. AUV trajectory obtained using GPS, INS sensors (dead reckoning method) and GPS/INS using EKF with adaptation.

of position obtained from dead reckoning method (using INS sensor data). Fig. 13 shows that the R matrix has been adjusted accordingly and more weight is given to the GPS data, and therefore the estimated trajectory in the integrated INS/GPS is "pulled" a little bit further to the GPS trajectory. However, big discrepancies can still be appreciated between the integrated INS/GPS estimate with respect to the GPS fixes. There are several explanations to this erratic behaviour. The first possibility is that it is caused by the poor level of accuracy of the low-cost GPS being used in this particular application. It is important to note that the proposed algorithm has detected a persistent high actual covariance ($\hat{C}_{lnn_k}$) for both the $\lambda(t)$ and $\varphi(t)$ throughout the trajectory. This results in insufficient weight being given to the GPS fixes in the FEKF and more on the position obtained by the dead reckoning. The second possibility is that the GPS receiver did not lock into a sufficient number of satellites

with a sufficiently small value of position dilution of precision that can provide the required level of accuracy. The use of a differential GPS receiver or a GPS receiver with a wide area augmentation system or a European Geostationary Navigation Overlay Service capability can be considered as a way forward to alleviate this problem.

5. Summary and conclusions

The problem with incomplete a priori knowledge of Q and R is considered. Within this paper, an adaptive Kalman filter approach, based on the filter innovation sequence coupled with fuzzy logic is discussed as an alternative for fusing INS sensor data and integrating INS/GPS position information. Implementation of this approach to the *Hammerhead* heading model, whose responses are measured with electronic compass, IMU and two additional sensors with different noise characteristics, has shown a promising result in improving the estimation of an individual SKF and EKF and enhancing the overall accuracy of the integrated INS/GPS.

Appendix A. Simple Kalman filter algorithms

Given a discrete-time controlled process described by the linear stochastic difference equations:

$$x_{k+1} = A_k x_k + B_k u_k + w_k, \tag{A.1}$$

$$z_k = H_k x_k + v_k, \tag{A.2}$$

where x_k is an $n \times 1$ system state vector, A_k is an $n \times n$ transition matrix, u_k is an $l \times 1$ vector of the input forcing function, B_k is an $n \times l$ matrix, w_k is an $n \times 1$ process noise vector, z_k is an $m \times 1$ measurement vector, H_k is an $m \times n$ measurement matrix and v_k is an $m \times 1$ measurement noise vector. Both the w_k and v_k are assumed to be uncorrelated zero-mean Gaussian white noise sequences with covariance given by

$$E[w_k w_i^T] = \begin{cases} Q_k, & i = k, \\ 0, & i \neq k, \end{cases} \tag{A.3}$$

$$E[v_k v_i^T] = \begin{cases} R_k, & i = k, \\ 0, & i \neq k, \end{cases} \tag{A.4}$$

$$E[w_k v_i^T] = 0, \text{ for all } k \text{ and } i. \tag{A.5}$$

The SKF algorithm can be organised into time update and measurement update equations

Time update equations:

$$\hat{x}_{k+1}^- = A_k \hat{x}_k + B_k u_k, \tag{A.6}$$

$$P_{k+1}^- = A_k P_k A_k^T + Q_k. \tag{A.7}$$

Measurement update equations:

$$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1}, \tag{A.8}$$

$$\hat{x}_k = \hat{x}_k^- + K_k[z_k - H_k \hat{x}_k^-], \tag{A.9}$$

$$P_k = [I - K_k H_k] P_k^-. \tag{A.10}$$

The measurement update equations incorporate a new observation into the a priori estimate from the time update equations to obtain an improved a posteriori estimate. In the time and measurement update equations, $\hat{x}_k$ is an estimate of the system state vector x_k, K_k is the Kalman gain and P_k is the covariance matrix of the state estimation error.

References

Asakawa, K., Kojima, J., Kato, Y., Matsumoto, S., & Kato, N. (2000). Autonomous underwater vehicle aqua explorer 2 for inspection of underwater cables. *Proceedings of the 2000 international symposium on underwater technology*, Tokyo, Japan, pp. 242–247.

Chaer, W. S., Bishop, R. H., & Ghough, J. A. (1997). Mixture-of-experts framework for adaptive Kalman filtering. *IEEE Transaction on Systems, Man and Cybernetics—Part B: Cybernatics, 27*(3), 452–464.

Escamilla-Ambrosio, P. J., & Mort, N. (2001). A hybrid Kalman filter-fuzzy logic multisensor data fusion architecture with fault tolerant characteristics. *Proceedings of the 2001 international conference on artificial intelligence*, Las Vegas, NV, USA, pp. 361–367.

Fitzgerald, R. J. (1971). Divergence of the Kalman filter. *IEEE Transaction on Automatic Control, AC-16*(6), 736–747.

Gade, K., & Jalving, B. (1999). An aided navigation post processing filter for detailed seabed mapping UUVs. *Modeling, Identificaton and Control, 20*(3), 165–176.

Grenon, G., An, P. E., Smith, S. M., & Healey, A. J. (2001). Enhancement of the inertial navigation system for the morpheous autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering, 26*(4), 548–560.

Hanlon, P. D., & Maybeck, P. S. (2000). Multiple-model adaptive estimation using a residual correlation Kalman filter bank. *IEEE Transaction on Aerospace and Electronic Systems, 36*(2), 393–406.

Jetto, L., Longhi, S., & Vitali, D. (1999). Localization of a wheeled mobile robot by sensor data fusion based on a fuzzy logic adapted Kalman filter. *Control Engineering Practice, 7*, 763–771.

Kailath, T. (1968a). An innovations approach to least-squares estimation, Part I: Linear filtering in additive noise. *IEEE Transaction on Automatic Control, AC-13*(6), 646–655.

Kailath, T. (1968b). An innovations approach to least-squares estimation, Part II: Linear smoothing in additive white noise. *IEEE Transaction on Automatic Control, AC-13*(6), 655–660.

Kailath, T. (1970). The innovations approach to detection and estimation theory. *IEEE Proceedings, 58*(5), 680–695.

Kinsey, J. C., & Whitcomb, L. L. (2003). Preliminary field experience with the DVLNAV integrated navigation system for manned and unmanned submersibles. *Proceedings of the first IFAC workshop on guidance and control of underwater vehicles*, Newport, South Wales, UK, pp. 83–88.

Kobayashi, K., Cheok, K. C., Watanabe, K., & Munekn, F. (1998). Accurate differential global positioning system via fuzzy logic Kalman filter sensor fusion technique. *IEEE Transactions on Industrial Electronics, 45*(3), 510–518.

Loebis, D., Chudley, J., & Sutton, R. (2003a). A fuzzy Kalman filter optimized using a genetic algorithm for accurate navigation of an autonomous underwater vehicle. *Proceedings of the sixth IFAC conference on manoeuvring and control of marine craft*, Girona, Spain, pp. 19–24.

Loebis, D., Sutton, R., & Chudley, J. (2002). Review of multisensor data fusion techniques and their application to autonomous underwater vehicle navigation. *Journal of Marine Engineering and Technology, A1*, 3–14.

Loebis, D., Sutton, R., & Chudley, J. (2003b). A fuzzy Kalman filter for accurate navigation of an autonomous underwater vehicle. *Proceedings of the first IFAC workshop on guidance and control of underwater vehicles*, Newport, South Wales, UK, pp. 161–166.

Magill, D. T. (1965). Optimal adaptive estimation of sampled stochastic processes. *IEEE Transaction on Automatic Control, AC-10*(4), 434–439.

Mehra, R. K. (1970). On the identification of variances and adaptive Kalman filtering. *IEEE Transactions on Automatic Control, AC-15*(2), 175–184.

Mehra, R. K. (1971). On-line identification of linear dynamic systems with applications to Kalman filtering. *IEEE Transaction on Automatic Control, AC-16*(1), 12–21.

Mohamed, A. H., & Schwarz, K. P. (1999). Adaptive Kalman filtering for INS/GPS. *Journal of Geodesy, 73*, 193–203.

Sterkersen, N., Kristensen, J., Indreeide, A., Selm, J., & Glancy, T. (1998). Hugin—UUV for seabed surveying. *Sea Technology, 39*(2), 99–104.

Wright, J., Scott, K., Tien-Hsin, C., Lau, B., Lathrop, J., & McCormick, J. (1996). Multi-sensor data fusion for seafloor mapping and ordnance location. *Proceedings of the 1996 symposium on autonomous underwater vehicle technology*, Monterey, CA, USA, pp. 167–175.

Yun, X., Bachmann, R. E., McGhee, R. B., Whalen, R. H., Roberts, R. L., Knapp, R. G., Healey A. J., & Zyda, M. J. (1999). Testing and evaluation of an integrated GPS/INS system for small AUV navigation. *IEEE Journal of Oceanic Engineering, 24*(3), 396–404.

SYSTEM IDENTIFICATION, MODELLING AND CONTROL OF AN AUTONOMOUS UNDERWATER VEHICLE

W. Naeem, R. Sutton and J. Chudley
{wnaeem, rsutton, jchudley}@plymouth.ac.uk

Marine and Industrial Dynamic Analysis Research Group
Department of Mechanical and Marine Engineering
University of Plymouth, PL4 8AA, UK

Abstract: Autonomy of an underwater vehicle is in the main attributable to the design of a suitable guidance and control system. Generally, the control system is developed and tested first in simulation studies using a model of the vessel to gain confidence in the approach being adopted. Therefore, plant identification is imperative to gain insight about the system. This paper is concerned with a practical system identification (SI) method in order to obtain a model of an autonomous underwater vehicle (AUV) using input output data as opposed to painstaking mathematical modelling techniques. A model of the AUV is developed using SI and tested in simulations using a linear quadratic Gaussian (LQG) controller. Modelling and simulation results are presented. *Copyright © 2003 IFAC*

Keywords: Underwater vehicles, system identification, modelling and LQG control.

1. INTRODUCTION

Underwater vehicles are playing a major role in the offshore industry, especially at places where human intervention is not possible. Its applications could range from subsea cable/pipeline tracking for inspection purposes to mine clearing operations. Normally, remotely operated vehicles (ROVs) are used for these purposes, but in cases such as deep sea exploration or surveillance missions, ROVs are not adequate due to the presence of an umbilical and the need for some kind of support platform. Thus, ROVs have limited capabilities in such type of missions. Autonomous underwater vehicles (AUVs), on the other hand, are not limited by the length of tethered cable and do not require any human operator or support platform for guidance. The performance of an AUV is limited by the onboard batteries that confine the mission duration. In addition, in contrast to ROVs, AUVs have an onboard integrated navigation, guidance and control (NGC) system. The navigation system provides information related to the target and the vehicle itself using onboard sensors such as inertial navigation system (INS), compass, pressure transducer etc. This information is fed to the guidance system which by utilising some guidance

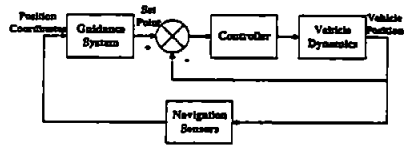


Fig. 1. Navigation, guidance and control of a vehicle

law generates reference headings. The control system is then responsible for keeping the vehicle on course as specified by the guidance system. A simple block diagram of an underwater vehicle NGC system is shown in Fig. 1.

This paper addresses the issue of designing and simulating a control system of an AUV by first developing a model using SI techniques from actual AUV trials data. A SI approach is useful in providing reliable and accurate models in a short time without relying too much on mathematical modelling techniques. This feature therefore, is attractive for the underwater vehicle manufacturers, where a vehicle configuration changes frequently to suit the mission requirements.

AUV modelling using SI approaches have been investigated before, (Tinker, *et al.*, 1979; Bertozzi, *et al.*, 2001; Goheen, and Jefferys, 1990; Ahmad, and Sutton, 2003), but most of the work involved has been done on identifying a model by generating data from a mathematical model of the vehicle. However, in this paper, the SI is performed on an actual AUV input output data obtained from trials explained in the next section. A discrete-time LQG controller is also developed and simulated for the identified AUV model explained in Section 3.

2. SYSTEM IDENTIFICATION

System identification of a dynamical system generally consists of the following four steps

1. Data acquisition
2. Characterisation
3. Identification/estimation
4. Verification

The first and most important step is to acquire the input/output data of the system to be identified. Acquiring data is not trivial and could be very much laborious and expensive. This involves careful planning of the inputs to be applied so that sufficient information about the system dynamics is obtained. If the inputs are not well designed, then it could lead to insufficient or even useless data.

The second step defines the structure of the system, for example, type and order of the differential equation relating the input to the output. This means selection of a suitable model structure, e.g. auto-regressive with exogenous input (ARX), auto-regressive moving average with exogenous input (ARMAX), Output error etc.

The third step is identification/estimation, which involves determining the numerical values of the structural parameters, which minimise the error between the system to be identified, and its model. Common estimation methods are least squares (LS), instrumental-variable (IV), maximum-likelihood (MLE) and the prediction-error method (PEM).

The final step, verification, consists of relating the system to the identified model responses in time or frequency domain to instil confidence in the obtained model. Residual (correlation) analysis, Bode plots and cross-validation tests are generally employed for model validation.

The above-mentioned features of system identification are symbolically indicated in Fig. 2.

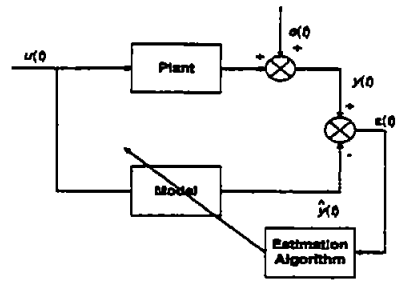


Fig. 2. The overall identification procedure

In this figure, u is the input, y is the output or response, e is the disturbance, $\hat{y}$ is the response of the model to the same input u and ε is the error between the model output and plant output also called the residuals. The objective of identification is to minimise the sum-squared errors or residuals ε. More details can be found in Ljung, (1999).

2.1 System description

The vehicle used in this study called *Hammerhead* has a torpedo shaped body about three and a half metres long and approximately one-third of a meter in diameter. The control surfaces are the two rear rudders for steering and two front hydroplanes for diving. The rudder and hydroplanes are controlled by two separate onboard stepper motors and the signal to the stepper motors is sent through an umbilical attached to the rear end of the vehicle. The onboard sensors include inertial navigation system (INS), TCM2 compass, pressure sensor, global positioning system (GPS), and a shaft speed encoder. The data logged using the above mentioned sensors is summarised below:

INS	heading, pitch, roll, linear and angular velocities
TCM2 Compass	heading, pitch and roll
Pressure sensor	depth of the vehicle
GPS	co-ordinates of the vehicle on the surface, forward speed
Shaft speed Encoder	vehicle speed

Fig. 3 depicts the sectional view of the *Hammerhead* AUV showing the hardware setup. The other end of the umbilical is attached to a control computer used to send and receive various signals. The rudder/ heading angle data pair is used to generate the yaw model while the hydroplane angle/depth is used to

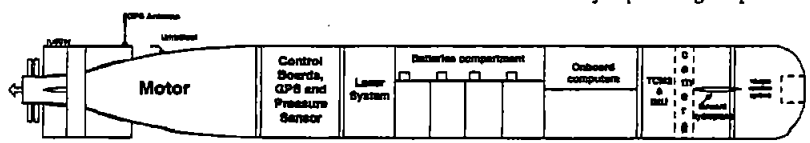


Fig. 3. Sectional view of the *Hammerhead* AUV

develop the depth channel model. Cross coupling effects between different channels such as yaw and roll of the vehicle are also examined.

3. EXPERIMENT DESIGN

The inputs applied to any system to be identified must be carefully designed prior to the experiment. This is imperative in obtaining good quality data that contains sufficient information about the system dynamics. Reckless design of inputs could lead to useless data containing very little or no information about the system behaviour. Some common types of input sequence are uniformly distributed random numbers, pseudo random binary sequence (PRBS), Fig. 4, and its variants such as multistep, Fig. 5, or doublet input, Fig. 6.

Each of the inputs was applied for a specified duration during which sensors data was acquired. The following algorithm shows the order in which identification data was obtained.

Step 1. *send input to the control surface*
Step 2. *while time < specified duration*
read depth sensor
read TCM2 compass
read IMU
end
Step 3. *read shaft speed*
read GPS
goto step 1.

With this configuration, the sampling frequency obtained was 8Hz. The data was resampled afterwards at 2Hz since this frequency was found adequate to control the *Hammerhead*. Moreover, it also helps smooth the data i.e. acts as a low pass filter. It was observed that during the transmission phase to the onboard actuators, no data could be acquired. This is due to the limitations of MATLAB. This problem was circumvented by leaving holes during that interval which represents the missing data. In addition, since there was no feedback from the control surfaces, the transition from one input to the other is approximated as a ramp and appropriate values are inserted. The whole input/output data was later processed and the missing data was interpolated. Fig. 7(a) shows data set with holes and Fig. 7(b) depicts the processed data.

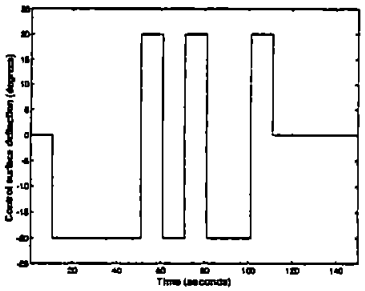


Fig. 4. First fifteen samples of a PRBS input for system identification

Most of the yaw channel identification trials were done on the surface therefore any combination of inputs could be applied to the rudder. However, for depth channel identification, it was found that the vehicle would hit the bed without the depth autopilot in the loop. Therefore, it has been decided to perform multiple experiments with a different multistep input, similar to Fig. 5, being applied to the hydroplane in each experiment. The experiments can easily be merged for model identification using the SI toolbox in MATLAB.

Once the model is identified, controllers can be designed and tested in simulation environment. There are several control systems available to be implemented on an AUV. A good account of various control laws for AUVs is documented by Craven *et al.* (1998). However, in this paper, a discrete-time LQG controller is developed and simulated and is explained in the next section.

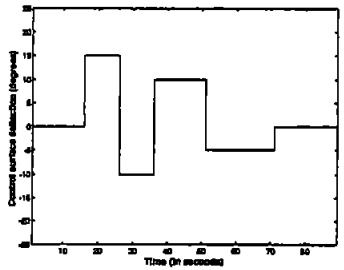


Fig. 5. Multistep input for system identification

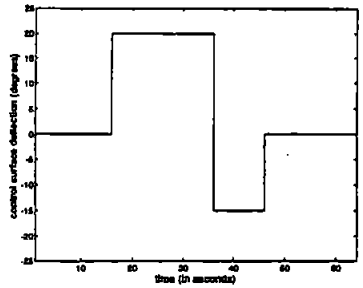


Fig. 6. Doublet input for system identification

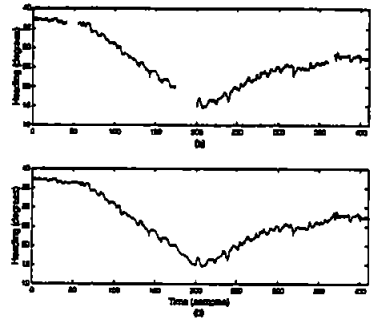


Fig. 7. (a) Original data set and (b) Interpolated data set.

4.LQG CONTROLLER

Recently optimal control theory has been extensively used to solve various control engineering problems, especially with the advent of powerful digital computers, the computation time is curtailed to a considerable extent. The optimal control is simply a minimisation or maximisation problem for which an objective function is defined that could involve different design parameters or states to optimise.

LQG is an optimal controller whose name is derived from the fact that it assumes a linear system, quadratic cost function and Gaussian noise. Unlike the pole placement method, where the designer must know the exact pole locations, LQG places the poles at some arbitrary points within the unit circle so that the resulting system is optimal in some sense. A linear quadratic state feedback regulator (LQR) problem is solved which assumes that all states are available for feedback. However, this is not always true because either there is no available sensor to measure that state or the measurement is very noisy. A Kalman filter can be designed to estimate the unmeasured states. The LQR and Kalman filter can be designed independently and then combined to form an LQG controller as shown in Fig. 8, a fact known as the *separation principle*.

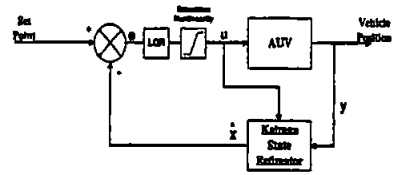


Fig. 8. LQG controller showing LQR gain and state estimator

5. RESULTS

The procedure detailed in Section 3 was used to acquire the input/output data from *Hammerhead* necessary for system identification. Trials have been performed at South West Water's Roadford reservoir, and at Willen Lake in Milton Keynes. The following subsections present system identification results applied to the test trials data and the resulting LQG control system as applied to the extracted model.

5.1 Identification results

As mentioned earlier, the yaw channel identification trials were performed on the surface using variants of PRBS input while short multistep input sequence was used for depth channel identification which are later merged for system identification. It was observed that that there is no coupling between the yaw and depth channels in this case so single-input single-output (SISO) models were developed. In addition, the pitch and roll does not seem to be effected with the change of heading, therefore the need for a SISO model is emphasised.

Rudder-yaw channel. One of the input sequence used in identification of rudder-yaw channel is a 32 length PRBS signal shown in Fig. 9(a). The response of *Hammerhead* to this input is also shown in Fig. 9(b). The data set is compared with the other heading responses with same input magnitude and found consistent. An ARX(221) model was extracted and the autocorrelation (ACF) and cross correlation functions (CCF) are depicted in Fig. 10. Although the CCF was slightly outside the 95% confidence band, the model was deemed adequate for further analysis. Cross validation test is performed to gauge the predicting capability of the model. In this test, data not used for SI is applied to the model and the simulated output is compared with the measured output. Two cross validation tests are performed for this channel and are shown in Figs. 11 and 12. The simulated outputs as seen from the figures matches reasonably well with the measured outputs.

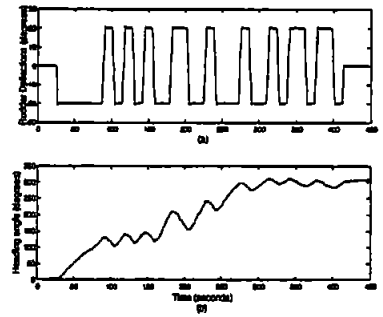


Fig. 9. (a) 32- length PRBS input sequence and (b) Hammerhead response to the PRBS

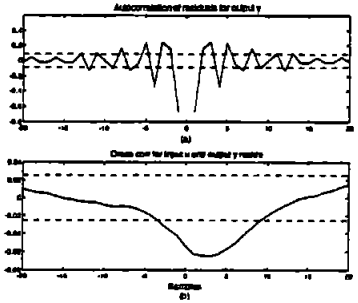


Fig. 10. (a) Autocorrelation of residuals for output and (b) Cross correlation of input u and output residuals.

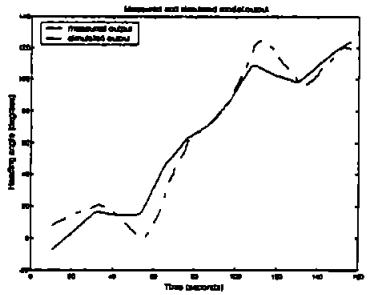


Fig. 11. First cross validation test for yaw channel model.

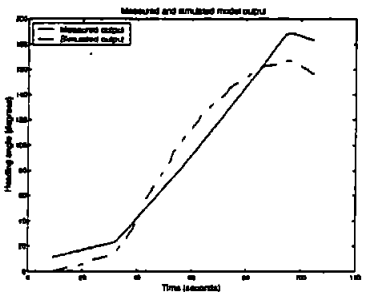


Fig. 12. Second cross validation test for yaw channel model

Sternplane-depth channel. As described in Section 3, short duration multistep inputs were applied to the hydroplane and data was collected. Unfortunately, after few runs, the umbilical used to send and receive the data was cut by the propeller and the mission had to be aborted. Since the data was inadequate to be used for SI, only heading controller was developed and the depth channel identification trials were postponed to a later date. In the following, the data that was collected during those runs is shown in Figs. 13 and 14.

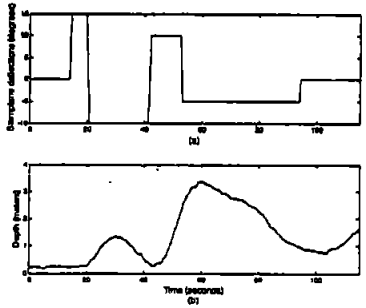


Fig. 13. Sternplane-depth channel identification trials data set 1 (a) Sternplane deflections in degrees and (b) Depth in meters.

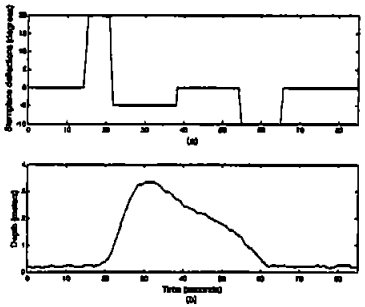


Fig. 14. Sternplane-depth channel identification trials data set 2 (a) Sternplane deflections in degrees and (b) Depth in meters.

Controller implementation. A discrete-time LQG controller was developed and applied to the extracted model as presented in Section 4. The task is to achieve a desired heading with minimum control effort and within the specified actuator constraints. The constraints on rudder actuator of *Hammerhead* are ±22 degrees. The LQG controller requires the tuning of four parameters which demands tedious trial and error procedure. To evade this problem, the tuning procedure of Maciejowski, (1985) has been adopted that requires the tuning of only two parameters and the other two are assigned values according to an automatic procedure. The controller was simulated and the results are depicted below. Fig. 15 shows the *Hammerhead* following a step change in reference heading and Fig. 16 presents the controller output. As shown, the rudder requires minimum control effort and it stays within the specified bounds.

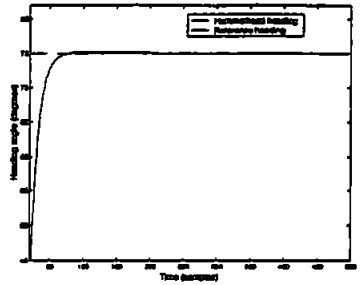


Fig. 15. LQG control of *Hammerhead* showing the yaw response to a step change in the heading.

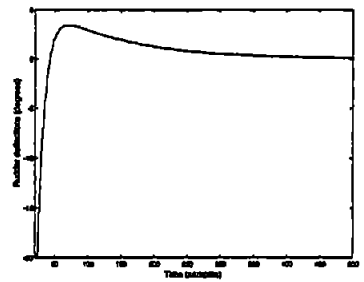


Fig. 16. Optimal rudder deflections generated by the LQG controller.

CONCLUSION

The design of an autopilot for an underwater vehicle generally requires a mathematical model of the physical system. Unfortunately, modelling an underwater vehicle is not trivial and could be very expensive and time consuming. This paper presents practical system identification using only the input/output data obtained from test trials without going into details of mathematical modelling involving horrendous differential equations. The complete hardware setup and the algorithm followed for acquiring data have been explained. The autopilot developed is an LQG controller for testing the modelling in simulation environment which will eventually be tested on the real system. Simulation results are presented to quantify the extracted model.

ACKNOWLEDGEMENTS

The authors would like to thank EPSRC for the funding of this project. Special thanks to the South West Water for providing access to their reservoir and to Willen Lake management for their help in conducting the initial surface trials.

REFERENCES

Ahmad, S. M. and R. Sutton (2003). Dynamic modelling of a remotely operated vehicle. In: *Proceedings of the 1st IFAC workshop on guidance and control of underwater vehicles GCUV'03*, April 9-11, pp. 47-52, Newport, South Wales, UK.

Bertozzi, C. G. Ippoliti, S. Radicioni and A. Rossolini (2001). Multiple models control of a remotely operated vehicle: Analysis of models structure and complexity. In: *Proceedings of the IFAC conference on control applications in marine systems, CAMS'2001*, Glasgow, Scotland, UK.

Craven, P. J., R. Sutton and R. S. Burns (1998). Control strategies for unmanned underwater vehicles. *The Journal of Navigation*, vol. 51, no. 1, pp. 79-105.

Franklin, G. F., J. D. Powell and M. Workman (1998). *Digital Control of Dynamic Systems, 3rd ed.* Addison-Wesley Longman Inc.

Goheen, K. R. and E. R. Jefferys (1990). The application of alternative modelling techniques to ROV dynamics. In: *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, May 13-18, pp. 1302-1309, Cincinnati, OH, USA.

Ljung, L. (1999). *System Identification, Theory for the User, 2nd ed.* Prentice Hall PTR.

Maciejowski, J. M. (1985). Asymptotic recovery for discrete-time systems. *IEEE Transactions on Automatic Control*, vol. AC-30, no. 6, pp. 602-605, June.

Tinker, S. J., A. R. Bowman and T. B. Booth (1979). Identifying submersible dynamics from free model experiments. In: *RINA annual report and transactions 1979*, pp. 191-196.

IFAC Publications
www.elsevier.com/locate/ifac

LQG/LTR Control of an Autonomous Underwater Vehicle Using a Hybrid Guidance Law

W. Naeem, R. Sutton and S. M. Ahmad
{w.naeem, r.sutton, s.ahmad}@plymouth.ac.uk

Marine and Industrial Dynamic Analysis Group
Department of Mechanical and Marine Engineering
The University of Plymouth, PL4 8AA, UK

Abstract: This paper addresses the issue of guidance and control of an autonomous underwater vehicle (AUV) for a cable tracking problem. A linear quadratic Gaussian controller with loop transfer recovery (LQG/LTR) is developed because of its strong robustness properties. The vehicle is guided towards the target using a combination of different guidance algorithms. The vehicle speed is used to formulate the guidance problem. Simulation results are presented and a comparison is made between fix and variable AUV speeds. *Copyright © 2003 IFAC*

Keywords: Guidance, control, LQG/LTR, cable tracking, autonomous underwater vehicle.

1. INTRODUCTION

Guidance and control of AUVs have seen a tremendous growth and development in the last few years and there have been significant applications of guidance and control systems for missions such as cable/pipeline tracking, mines clearing operation, deep sea exploration, feature tracking etc. For an AUV to work effectively, a well-integrated navigation, guidance and control (NGC) system is imperative. A simple block diagram of an NGC system is depicted in Figure 1. The navigation system generates information about the vehicle position, velocity, heading etc. using various onboard sensors such as a compass, global positioning system (GPS), pressure sensor etc. The guidance system manipulates the navigation information and generates suitable references to be followed by the AUV. The control system is then responsible for keeping the vehicle on course as specified by the guidance system.

The main difference between an AUV and ROV (remotely operated vehicle) is that the ROV is controlled by a trained human operator while the AUV is steered by an onboard guidance system. In

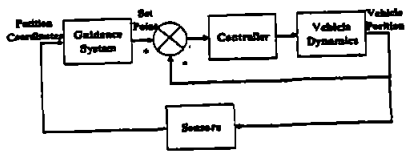


Fig. 1. Navigation, guidance and control of a vehicle

this regard, the guidance system plays the key role in bringing autonomy to the vehicle. The purpose of this paper is to develop an integrated guidance and control algorithm for an AUV test model, which will eventually be developed and tested in real time in an actual AUV.

A plethora of control systems is available to be implemented on an AUV. A good account of various control systems is presented by Craven (1999), while Naeem *et al.* (2003), recently documented a review on various guidance laws for underwater vehicles.

Todate optimal control theory has been extensively used to solve various control engineering problems. Especially with the advent of powerful digital computers, the computation time is curtailed to a considerable extent. The optimal control is simply a minimisation or maximisation problem for which an objective function is defined that could involve different design parameters or states to optimise.

Linear quadratic Gaussian (LQG) is an optimal controller whose name is derived from the fact that it assumes a linear system, quadratic cost function and Gaussian noise. Unlike pole placement method, where the designer must know the exact pole locations, LQG places the poles at some arbitrary points within the unit circle so that the resulting system is optimal in some sense. A linear quadratic state feedback regulator (LQR) problem is solved which assumes that all states are available for feedback. However, this is not always true because either there is no available sensor to measure that state or the measurement is very noisy. A Kalman filter can be designed to estimate the unmeasured states. The LQR and Kalman filter can be designed independently and then combined to form an LQG controller, a fact known as the *separation principle*. Individually the LQR and Kalman filter have strong robustness properties with gain margin up to infinity and over 60° phase margin, (Burl, 1999). Unfortunately, the LQG has relatively poor stability margins which can be circumvented by using loop transfer recovery (LTR). A discrete time LQG/LTR design is presented in this paper motivated from the work of Maciejowski (1985). The LTR works by adding fictitious noise to the process input which effectively cancels some of the plant zeros and possibly some of the stable poles, and inserts the estimator's zeros (Maciejowski, 1985; Skogestad and Postlethwaite, 1996). The hybrid guidance law developed utilises vehicle speed as a means to formulate the guidance problem that was first proposed by Naeem *et al.*, (2003) and is simulated in this paper.

The paper is organised as follows. The next section describes the AUV model while Section 3 explains the LQG/LTR control system design. Section 4 states the guidance law formulation and simulation results are presented in Section 5. Finally, concluding remarks are made in Section 6.

2. AUV MODELLING

The AUV test model is that used by Kwiesielewicz *et al.* (2001), where the model parameters are given in terms of vehicle speed. Since the guidance law require different vehicle speeds, therefore this model has been chosen for demonstration of the proposed algorithm. The single-input single-output (SISO) vehicle model for a given vehicle speed can be described by the following transfer function,

$$G(s) = \frac{as + b}{s(s^2 + cs + d)} \qquad (1)$$

where the coefficients a, b, c and d are given in terms of vehicle speed v in knots.

$$a = 0.05803\,v^2,\ b = 0.00449\,v^3$$
$$c = 0.25963\,v,\ d = 0.00856\,v^2$$

The input to the AUV are the rudder deflections while the output is the heading of the vehicle. The model parameters are calculated for three different vehicle speeds and the resulting continuous time model is discretised at a sampling rate of 10 Hz. The discretized models are then converted into state space controllable canonical forms owing to the LQG controller requirements. The LQG/LTR controller requires the model to be minimum phase and should be controllable and observable. The model is tested for these requirements and found to be good for said purposes. The model AUV is assumed to have a turning radius of 25 m and the constraints on the rudder actuator are maximum 25 degrees in either left or right direction.

3. CONTROL SYSTEM DESIGN

LQG/LTR control of an unmanned underwater vehicle (UUV) has been reported by Juul *et al.*, (1994), and Triantafyllou and Grosenbaugh, (1991). However, both these papers deal with multivariable continuous LQG/LTR control of an underwater vehicle assuming that the guidance commands are available. In this paper, a discrete-time LQG/LTR controller is developed which is more realistic for practical purposes. A guidance law is also developed which generates suitable commands to be followed by the vehicle and is the subject of next section. In the following, the LQG/LTR controller is developed for the AUV model shown in Section 2.

3.1 Design Specifications

Two types of design specifications are usually given prior to any controller design which are closely related. The time domain specifications involve the maximum overshoot, settling time etc. while the frequency domain specifications provides the bandwidth, gain margin (GM), phase margin (PM) etc. of the system. They can be evaluated by generating the step response and Bode plot of the system respectively. However, in an LQG/LTR design, frequency tuning is usually desired. A Bode plot of the open loop system (Equation 1), suggests infinite gain at zero frequency therefore, *gain crossover frequency* (gcf) is used as a measure of the bandwidth of the system. The desired gcf of the open loop system for all vehicle speeds is 1 Hz. An acceptable nominal design usually is one that attains both a GM ≥ 3 dB and PM ≥ 30°, (Wolovich, 1994). The desired GM in this case is set at 10 dB while the PM at 53°, well above the nominal values.

3.2 Kalman Filter Design

Since the heading of the AUV corrupted by noise is the only measured variable, the remaining states have to be measured through a state estimator prior to

control calculations. A current estimator is used because the estimate is based on the current measurement. This is done because the processing time required to compute each control signal is small in contrast to the sampling time. In addition, this scheme gives more accurate results as compared to a prediction estimator (Franklin *et al.*, 1998). Let the plant to be controlled is modelled in state space form as

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k)$$
$$y(k) = \mathbf{C}x(k) \tag{2}$$

The design objective is to find the Kalman gain $\mathbf{K}_f$ such that the estimate of $\mathbf{x}(k)$ is optimal. The solution to this problem is given by the discrete steady state Kalman filter gain equation given by (Franklin *et al.*, 1998; Maciejowski, 1985)

$$\mathbf{K}_f = \mathbf{P}\mathbf{C}^T(\mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{V})^{-1} \tag{3}$$

where $\mathbf{V}$ is the measurement noise spectral density matrix and $\mathbf{P}$ is the steady state error covariance matrix given by the solution of a discrete steady state Riccatti equation, (Maciejowski, 1985)

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^T - \mathbf{A}\mathbf{P}\mathbf{C}^T(\mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{V})^{-1}\mathbf{C}\mathbf{P}\mathbf{A}^T + \mathbf{W} \tag{4}$$

where $\mathbf{W}$ is the process noise spectral density matrix. The parameters $\mathbf{W}$ and $\mathbf{V}$ are tuned until the desired filter's open-loop return ratio $\Phi(z)$ specifications are met which is shown below

$$\Phi(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{A} * \mathbf{K}_f \tag{5}$$

3.3 LQR Design

Once the Kalman gain is evaluated for the desired specifications for all models, the LQR state feedback gains are calculated. An objective function is minimised given by

$$J = \frac{1}{2}\sum_{k=0}^{N}\left[\mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + u^T(k)\mathbf{R}u(k)\right] \tag{6}$$

where the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ are chosen according to Maciejowski (1985) as

$$\mathbf{Q} = \mathbf{C}^T\mathbf{C}, \quad \mathbf{R} \approx 0 \tag{7}$$

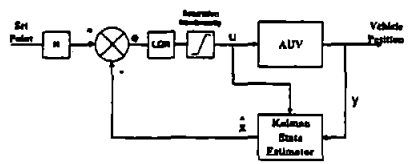


Fig. 2. LQG controller showing LQR gain and state estimator

The above values provide asymptotic recovery of the stability margins, given that the plant obeys some specific characteristics. The state feedback matrix $\mathbf{K}_c$ is obtained by solving equations dual to Equations 3 and 4, and is used to generate the control according to

$$u(k) = -\mathbf{K}_c\hat{\mathbf{x}}(k) \tag{8}$$

where $\hat{\mathbf{x}}$ is the estimate of the state $\mathbf{x}$ given by Equation 2, and u is the control action. The closed form solution of $\mathbf{K}_c$ for the values of $\mathbf{Q}$ and $\mathbf{R}$ in Equation 7, is given by Maciejowski (1985) as

$$\mathbf{K}_c = (\mathbf{C}\mathbf{B})^{-1}\mathbf{C}\mathbf{A} \tag{9}$$

A feedback compensator is finally synthesised as a series connection of the Kalman filter and the optimal state-feedback controller as depicted in Figure 2 given by (Maciejowski, 1985)

$$\hat{\mathbf{x}}(k+1) = (\mathbf{A} - \mathbf{B}\mathbf{K}_c - \mathbf{K}_f\mathbf{C}\mathbf{A} + \mathbf{K}_f\mathbf{C}\mathbf{B}\mathbf{K}_c)\hat{\mathbf{x}}(k) + \cdots (\mathbf{A} - \mathbf{B}\mathbf{K}_c)\mathbf{K}_f e(k)$$
$$u(k) = \mathbf{K}_c(\mathbf{I} - \mathbf{K}_f\mathbf{C})\hat{\mathbf{x}}(k) + \mathbf{K}_c\mathbf{K}_f e(k) \tag{10}$$

where $e = r - x$, is the error between a reference signal r and desired state x. Let $\mathbf{G}(z)$ is the transfer function of the system defined by Equation 2 and $\mathbf{H}(z)$ is the compensator transfer function. If the plant $\mathbf{G}(z)$ is minimum phase and $\det(\mathbf{CB}) \neq 0$, then full recovery is achieved if

$$\mathbf{G}(z)\mathbf{H}(z) = \Phi(z) \tag{11}$$

where $\mathbf{G}(z)\mathbf{H}(z)$ is called the loop transfer function.

4. GUIDANCE LAW

The objective of any guidance law is to steer the AUV so that it intercepts the target in minimum time and maximum accuracy. The guidance law used in this paper utilises AUV speed as a means to formulate the problem. The complete mission is classified into four different phases utilising different guidance laws. These are i) launch phase, ii) midcourse phase, iii) terminal phase, and iv) tracking phase as shown in Figure 3. In the first phase called the launch phase or the boost phase, the vehicle is launched from a vessel and guided in the direction of the line of sight (LOS) with maximum speed, using the LOS guidance only. Once the vehicle approaches the LOS, midcourse guidance could be invoked. In midcourse phase, the vehicle follows the LOS angle with maximum speed using the way point guidance, (Healey and Lienard, 1993). During this part of the flight, changes may be required to bring the vehicle onto the desired course and to make certain that it stays on that course. The midcourse guidance system is used to place the vehicle near the target area, where the system to be used in the final phase of guidance can take over. It should be noted that there is no need for the vehicle to submerge at this stage, as the objective is to approach the target area with maximum accuracy regardless of the orientation of

the vehicle with respect to the cable. When the vehicle reaches within the circle of acceptance, the third phase called the terminal phase is invoked. During this phase the vehicle must be slowed down and submerged in order to line up with the cable/pipeline as shown in Figure 3. The circle of acceptance in this case as opposed to Healey and Lienard (1993), should be taken at least the minimum turning radius of the vehicle in order to avoid overshoot. Finally, when the vehicle enters the waypoint, the fourth phase called the tracking phase is called up utilising any existing guidance law with the vehicle speed reduced to its minimum value. For example, the vehicle could use vision based guidance system to follow the cable. If the cable to be followed is an electrical/ communication cable, then magnetometers could be used to detect the radiation from the cable and guide the vehicle in the appropriate direction (Naeem *et al.*, 2003).

To implement the guidance law, it is necessary to compute the LOS angle λ. This requires relative positions of the AUV and target in both the co-ordinates i.e.,

$$h = y_t - y_v$$
$$r = x_t - x_v$$

therefore,

$$\lambda = \tan^{-1}\left(\frac{h}{r}\right)$$

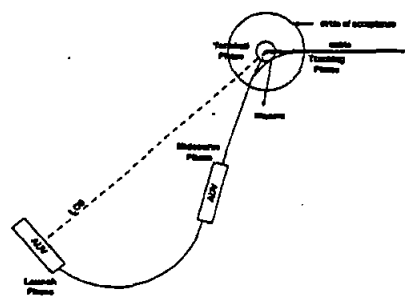


Fig. 3. Planar view of the four phases of flight for cable tracking problem of an AUV.

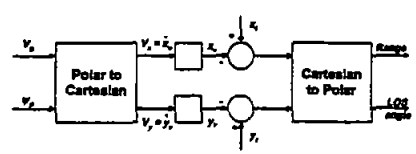


Fig. 4. Guidance subsystem block diagram

The kinematic equations of the AUV are stated below and represents the components of the velocity in the (x, y) plane

$$V_x = V_p\cos\psi_p$$
$$V_y = V_p\sin\psi_p$$

where ψ_p and V_p represents the actual heading and velocity of the AUV respectively. The speed of the AUV is regulated at three different values used in different phases of the mission as mentioned above. The components (x_v, y_v) of the AUV position can be evaluated by integrating the velocities (V_x, V_y), respectively. In addition to the LOS angle from the vehicle to the target, the guidance system also generates the range (distance) of the AUV from the target. The range measure is used to switch between different pre-tuned controllers. The guidance subsystem block diagram is shown in Figure 4 and is implemented in Simulink environment.

5. SIMULATION RESULTS

The proposed integrated guidance and control algorithm is implemented on the AUV model shown in Section 2 in Matlab/Simulink environment. The following assumptions are taken for the simulations:

i) The AUV and target are in the same plane.
ii) Complete navigational information is available through onboard sensors.
iii) A complete knowledge of the target's motion is available to the AUV.
iv) The AUV is equipped with a vision system that generates the co-ordinates of the points on the cable to be tracked.
v) The initial target co-ordinates (one end of the cable) are known prior to the mission.

The first step in any LQG/LTR control problem is to design a target filter's open-loop return ratio given by Equation 5, which requires the Kalman gain to be evaluated. By manipulating the spectral density matrices $\mathbf{W}$ and $\mathbf{V}$ in Equations 3 and 4, the Kalman filter can be designed and hence the target filter's open-loop return ratio. In this paper, the procedure adopted by Weerasooriya and Phan (1995) is followed. Keeping the measurement noise spectral density fixed at unity and tuning the process noise spectral density matrix, until the desired frequency domain specifications are met. The Bode plot of the desired filter's open-loop return ratio for the 10 knots speed model of the vehicle is shown in Figure 5. The GM, PM and gcf can be readily evaluated from the plot. The next step is to calculate the feedback gains using the optimal $\mathbf{Q}$ and $\mathbf{R}$ in Equation 7 and develop the LQG/LTR compensator using Equation 10. The loop transfer function $\mathbf{G}(z)\mathbf{H}(z)$ is also evaluated and the Bode plot superimposed on the Bode plot of the open-loop return ratio in Figure 6 shows the amount of recovery achieved. In this case, full recovery is

achieved as the two plots overlap each other. Figure 7 presents the step response of the closed loop feedback system showing a large overshoot. This can be reduced by adding more damping to the system by introducing a weighting factor on the diagonal term of Q corresponding to the velocity state. This is equivalent to using rate feedback for improving damping from a conventional sense (Weerasooriya and Phan, 1995). Figure 8 depicts the step response of the closed loop system with modified Q and Figure 9 presents the Bode plot of the loop transfer function. Although overshoot has subsided but at the cost of reduced stability margins due to the deviation from the optimal values. The same procedure has been applied to all vehicle models at various speeds and compensators are developed. Finally guidance and control system integration is done and the simulation results are shown in Figure 10 for a cable tracking mission, which clearly shows good tracking behaviour using the proposed guidance algorithm. The control surface deflections generated by the controller is depicted in Figures 11 and 12 for the case of optimal and modified Q respectively. Clearly, the modified Q with additional damping causes less variation in the control input as compared to optimal Q but at the cost of reduced stability margins. However, both figures suggest that the deflections are within the constrained actuator limits.

6. CONCLUSION

This paper demonstrates an integrated guidance and control system approach using an LQG/LTR controller and a hybrid guidance law. The LQR/LTR controller is synthesised in discrete-time and a hybrid guidance law is proposed which uses different vehicle speeds in different phases of the mission. Simulation results are presented to show the robustness properties of the proposed integrated system. Results for a cable-following mission also depicts good tracking performance. A SISO system is used for the simulations, however, a multivariable LQG/LTR integrated with the proposed guidance system is an area of ongoing research.

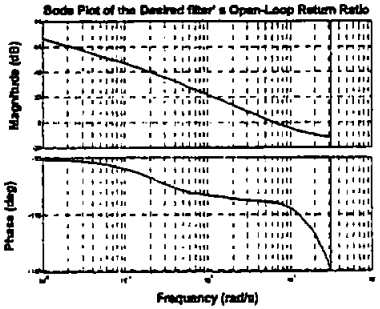


Fig. 5. Bode plot of the target's filter open-loop return ratio

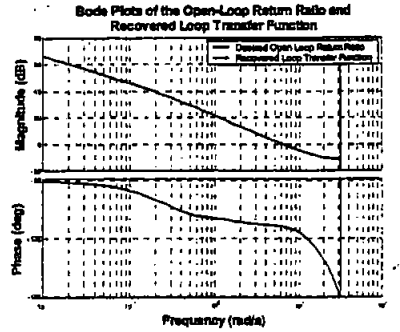


Fig. 6. Bode plots of the filter's open-loop return ratio and recovered loop transfer function for nominal Q (full recovery)

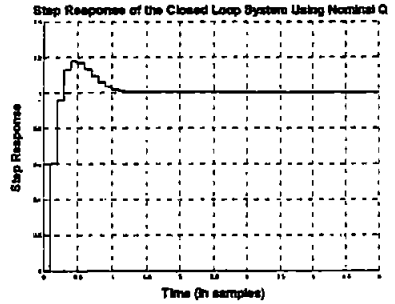


Fig. 7. Step response of the closed loop system for $Q=C^TC$ and $R=0$.

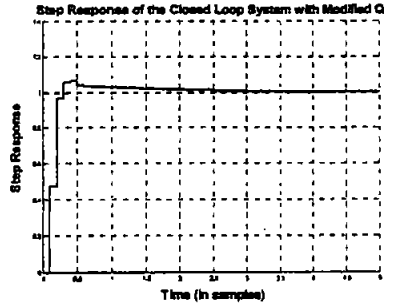


Fig. 8. Step response of the closed loop system for modified Q & R=0

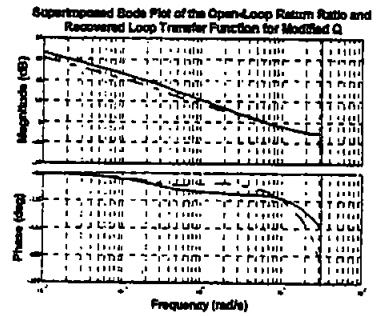


Fig. 9. Bode plots of the filter's open-loop return ratio (solid line) and recovered loop transfer function (dashed line) with added damping, (reduced stability margins)

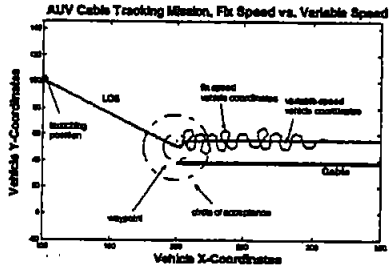


Fig. 10. Cable tracking mission from launching to tracking, variable speed vs. fixed speed

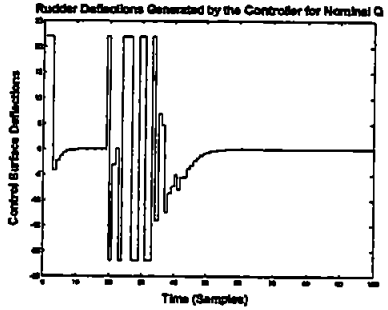


Fig. 11. Rudder deflections generated by the LQG/LTR controller for $Q=C^TC$ & $R=0$

REFERENCES

Burl, J. B., (1999). *Linear Optimal Control, H_2 and H_∞ Methods.* Addison-Wesley Longman Inc.

Craven, P. J. (1999). Intelligent control strategies for an autonomous underwater vehicle. *PhD Thesis, University of Plymouth, UK.*

Franklin, G. F., J. D. Powell and M. Workman (1998). *Digital Control of Dynamic Systems, 3rd ed.* Addison-Wesley Longman Inc.

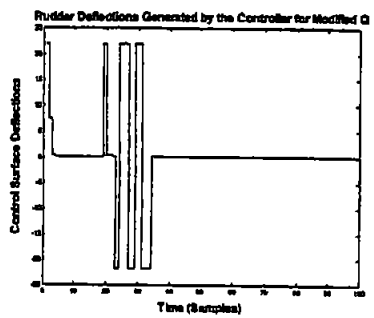


Fig. 12. Rudder deflections generated by the LQG/LTR controller for modified Q & R=0

Healey, A. J., and D. Lienard, (1993). Multivariable Sliding Model Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles. *IEEE Journal of Oceanic Engineering,* vol. 18, no. 3, pp. 327-339, July.

Juul, D. L., M. McDermott, E. L. Nelson, D. M. Barnett and G. N. Williams (1994). Submersible Control Using the Linear Quadratic Gaussian with Loop Transfer Recovery Method. In: *Proceedings of the Symposium on Autonomous Underwater Technology,* pp. 417-425, July, Cambridge, MA, USA.

Kwiesielewicz, M., W. Piotrowski and R. Sutton (2001). Predictive Versus Fuzzy Control of Autonomous Underwater Vehicle. *IEEE International Conference on Methods and Models in Automation and Robotics,* pp 609-612, 28-31 August, Miedzyzdroje, Poland.

Maciejowski, J. M., (1985). Asymptotic Recovery for Discrete-Time Systems. *IEEE Transactions on Automatic Control,* vol. AC-30, no. 6, pp. 602-605, June.

Naeem, W., R. Sutton, S. M. Ahmad, and R. S. Burns, (2003). A Review of Guidance Laws Applicable to Unmanned Underwater Vehicles. To be published in *The Journal of Navigation.* Vol. 56, no. 00, pp 1-15.

Skogestad, S. and I. Postlethwaite, (1996). *Multivariable Feedback Control: Analysis and Design using Frequency-domain Methods.* John Wiley and Sons Ltd.

Triantafyllou, M. S., and M. A. Grosenbaugh, (1991). Robust Control for Underwater Vehicle Systems with Time Delays. *IEEE Journal of Oceanic Engineering,* vol. 16, no. 1, pp 146-151, January.

Weerasooriya, S. and D. T. Phan, (1995). Discrete-Time LQG/LTR Design and Modelling of a Disk Drive Actuator Tracking Servo System. *IEEE Transactions on Industrial Electronics,* vol. 42, no. 3, pp 240-247,June.

Wolovich, W. A., (1994). *Automatic Control Systems, Basic Analysis and Design.* International Edition, Saunders College Publishing.

MODEL PREDICTIVE CONTROL OF AN AUTONOMOUS UNDERWATER VEHICLE

Wasif Naeem

Marine and Industrial Dynamic Analysis Group
Department of Mechanical and Marine Engineering
The University of Plymouth, PL4 8AA, UK

Abstract: This paper investigates the application of model predictive control to the yaw angle of an autonomous underwater vehicle. A simple line of sight guidance scheme is utilised to generate the reference heading, which is to be followed. Simulation results are presented to demonstrate the suitability of the proposed algorithm. *Copyright © 2002 IFAC*

Keywords: Model predictive control, autonomous underwater vehicle, guidance, genetic algorithm, navigation.

1. INTRODUCTION

Autonomous underwater vehicles (AUVs) are no longer engineering curiosities. They have been under development for over three decades and in the last few years there have been significant advances towards their use in operational missions (Millard, *et al.*, 1998). Although remotely operated vehicles (ROVs) play an important role in the offshore industry, their operational effectiveness is limited by the tethered cable and the reliance and cost of some kind of support platform. Given these limitations, developments in advance control engineering theory and the computation hardware for analysis, design and implementation, interest in the viability of employing AUVs in operational missions has been revived. The use of AUVs is increasingly being considered for applications such as cable/pipeline tracking, mines clearing operations, deep-sea exploration, feature tracking etc. The potential usage of AUVs is restricted by two main factors. The first is the limitation of battery power, which confines the AUVs for long duration missions. Most of the vehicles in use, uses car batteries that need to be recharged every few hours and that makes them unsuitable for long duration missions. The second limiting factor is associated with the current generation of onboard navigation, guidance and control (NGC) systems. The vehicle must have a reliable and well-integrated NGC system of which control is the key element responsible for keeping the vehicle on course. On the other hand, navigation

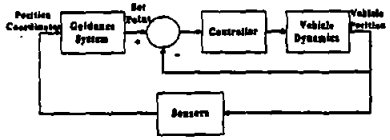


Fig. 1. Navigation, guidance and control of a vehicle

systems provide information related to the target and vehicle itself, using on-board sensors such as inertial navigation system (INS), compass, pressure transducer etc. This information is fed to the guidance system which by utilising some guidance law generate reference trajectories. A simple block diagram of the navigation, guidance and control system is depicted in Fig. 1.

Several classical and advance control strategies have been simulated and tested on an AUV, Craven (1999). In this paper, a new control strategy using model predictive control (MPC) is developed and simulated on an AUV to track the reference heading provided by a guidance system. MPC is chosen because of several reasons one of which, is the ability to handle constraints in a natural way. A genetic algorithm (GA) is used as an optimisation tool to aid in designing the controller which is motivated from the work of Duwaish and Naeem (2001).

The paper is organised as follows. Section 2 explores model predictive control and genetic algorithms. Problem formulation is described in Section 3 while Section 4 presents the simulation results. Finally, concluding remarks are given in Section 4.

2. MODEL PREDICTIVE CONTROL

MPC refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimise the future behaviour of a plant. Originally developed to meet the specialised control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace, metallurgy, and pulp and paper, (Qin and Badgewell, 1997).

The development of MPC can be traced back to 1978 after the publication of the paper by Richalet *et al.* Then Cutler and Ramaker from Shell Oil in 1979, 1980 developed their own independent MPC technology "Dynamic Matrix Control". The most popular form of predictive control called the generalized predictive control has been devised by Clarke *et al.*, (1987a, b) and is employed in this paper.

The process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the output of the process can be used. Further if the process is subject to disturbances, a disturbance or noise model can be added to the process model. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used. Typically the criterion is the difference between the predicted process output and the desired reference trajectory. A simple criterion function is

$$J = \sum_{i=1}^{H_p} [\hat{y}(k+i) - w(k+i)]^2 \qquad (1)$$

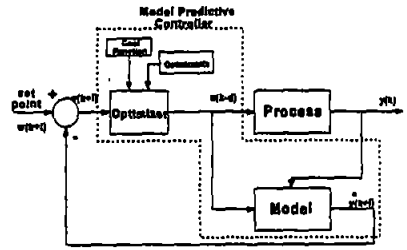


Fig. 2. Structure of Model Predictive Control

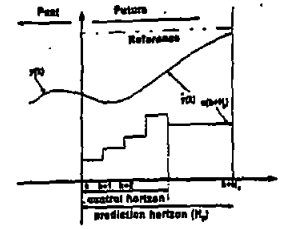


Fig. 3. Predicted output and the corresponding optimum input over a horizon H_p, where $u(k)$, optimum input, $\hat{y}(k)$, predicted output, and $y(k)$, process output.

where $\hat{y}$ is the predicted process output, w is the reference trajectory, and H_p is the prediction horizon or output horizon. The structure of an MPC is shown in Fig. 2.

The controller output sequence u_{opt} over the prediction horizon is obtained by minimisation of J with respect to u. As a result the future tracking error is minimised. If there is no model mismatch i.e. the model is identical to the process and there are no disturbances and constraints, the process will track the reference trajectory exactly on the sampling instants.

Model Predictive Control algorithm, consists of the following three steps.

i) Explicit use of a model to predict the process output along a future time horizon (Prediction Horizon).
ii) Calculation of a control sequence along a future time horizon (Control Horizon), to optimise a performance index.
iii) A receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated as shown in Fig. 3.

The selection of MPC to control an AUV is attributed to several factors. Some of them are listed below.

- The concept is equally applicable to single-input, single-output (SISO) as well as multi-input, multi-output systems (MIMO).
- MPC can be applied to linear and nonlinear systems.
- It can handle constraints in a systematic way during the controller design.
- The controller is designed at every sampling instant so disturbances can easily be dealt with.

In this paper, the optimisation of the performance index is done using GA, which is described in the next section.

2.1. Genetic Algorithms

GAs inspired by Darwinian theory, are powerful non-deterministic iterative search heuristics. GAs operate on a population consists of encoded strings, each string represents a solution. Crossover operator is used on these strings to obtain the new solutions, which inherits the good and bad properties of their parent solutions. Each solution has a fitness value, solutions having higher fitness values are most likely to survive for the next generation. Mutation operator is applied to produce new characteristics, which are not present in the parent solutions. The whole procedure is repeated until no further improvement is observed or run time exceeds to some threshold, (Sait and Youssef, 1999). The flowchart of a simple genetic algorithm is presented in Fig. 4 and the operation of the GA is explained as follows.

To start the optimization, GA use randomly produced initial solutions. This method is preferred when a priori knowledge about the problem is not available. After randomly generating the initial population of say N solutions, the GA use the three genetic operators to yield N new solutions at each iteration. In the selection operation, each solution of the current population is evaluated by it's fitness normally represented by the value of some objective function and individuals with higher fitness value are selected. Different selection methods such as roulette wheel selection and stochastic universal sampling can be used. The crossover operator works on pairs of selected solutions with certain crossover rate. The crossover rate is defined as the probability of applying crossover to a pair of selected solutions. There are many ways of defining this operator such as single point crossover, double point crossover, multi-point crossover etc. For example the single point crossover works on a binary string by determining a point randomly in the two strings and corresponding bits are swapped to generate two new solutions.

Mutation is a random alteration with small probability of the binary value of a string position. This operator prevents GA from being trapped in local minima. The fitness evaluation unit in GA acts as an interface between the GA and the optimization problem. Information generated by this unit about the quality of different solutions is used by the selection operation in the GA. Next the stopping criteria must be decided. This may be the case when there is no significant improvement in maximum fitness or the maximum allowable time (number of iterations) is passed. At the end of the algorithm, the best solution found so far is returned.

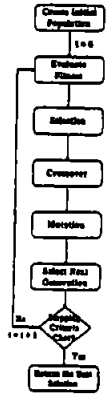


Fig. 4. Flowchart of a simple genetic algorithm

3. PROBLEM FORMULATION

In order to formulate the problem the following assumptions are made.

i) The AUV and target are assumed to be in the same plane.
ii) The target is assumed to be stationary, however, non-stationary targets can be easily dealt with and is area of active research.
iii) Navigation information is completely available to the guidance system.

The AUV-target engagement geometry is shown in Fig. 5. Both target and AUV are assumed to be point masses having co-ordinates (x_v, y_v) and (x_t, y_t) respectively. The guidance system generates the reference heading to be followed by the AUV which is simply the line of sight (LOS) angle λ formed between the AUV and the target given by Equation 2.

$$\lambda = \tan^{-1}\left(\frac{y_t - y_v}{x_t - x_v}\right) \qquad (2)$$

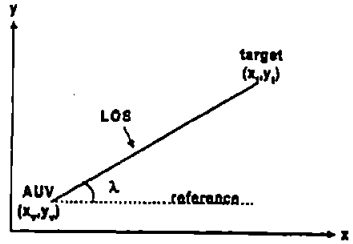


Fig. 5. AUV-target engagement geometry

The MPC is then responsible to direct the AUV towards the LOS. The following steps describe the operation of the MPC algorithm using GA. At any time step k

i) Evaluate process outputs using the process model.
ii) Use GA search to find the optimal control moves which optimise the cost function and satisfies process constraints. This can be accomplished as follows.
   (a) generate a set of random possible control moves. The control moves or population consists of real values which is reasonable in a real world environment.
   (b) find the corresponding process outputs for all possible control moves using the process models.
   (c) evaluate the fitness of each solution using the cost function and the process constraints.
   (d) apply the genetic operators (selection, crossover and mutation) to produce new generation of possible solutions.
   (e) repeat until predefined number of generations has reached and thus the optimal control moves are determined.
iii) Apply the optimal control moves generated in step 2 to the process.

4. SIMULATION RESULTS

The proposed MPC algorithm has been applied to an AUV test model supplied by the QinetiQ. The model relates the yaw angle to the rudder deflection. Dimensionally, the AUV is 7 meters long, 1 meter in diameter and has a displacement of 3600 kilograms. A full description of the equations of motion describing the dynamic behaviour of the vehicle in the lateral plane can be found in Craven, (1999). The model is derived from first principles i.e., using law of physics, which utilises theory of rigid body motion, kinematics and hydrodynamics. A simplified linear model is extracted using system identification from the non-linear MATLAB/Simulink simulation model provided. The identified model is of the form:

$$\dot{x} = Ax + Bu \qquad (3)$$

where A and B are the state and input matrices respectively. More precisely the two-dimensional core state-space model is given by,

$$\begin{bmatrix} \dot{\psi} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \psi \\ r \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} u$$

where ψ is the yaw or heading angle in degrees, r is the yaw rate in degrees/sec and u the input signal to the canard. The model is discretised at a sampling rate of 0.1 samples/sec and converted to a transfer function model for the generalised predictive control algorithm.

The rudder actuator can move a maximum of 25° to either left or right direction. The AUV initial position co-ordinates are (0,0) while the target is located at (200,0) giving the LOS angle λ equal to zero using Equation 2.

The MPC and GA parameters used in the simulation are provided in Table 1. The cost function used in this paper is given by

$$J = \sum_{j=1}^{H_p} e(k+j)^T Q e(k+j) + \sum_{j=1}^{H_c} \Delta u(k+j)^T R \Delta u(k+j) \qquad (4)$$

subject to

$$u^l \le u(k+j) \le u^u$$

where the superscripts l and u represents the lower and upper bounds on the input moves respectively. R is the weight on the rate of change of control moves and Q is the weight on the prediction error

$$e(k) = \hat{y}(k) - w(k)$$

The second term in Equation 4 represents the penalty on the rate of change of control moves. This is augmented to prevent excessive movements of the rudder.

The AUV heading when the MPC is applied is depicted in Fig. 6, which clearly shows that the AUV is following the LOS closely. Fig. 7 presents the controller output, which are actually the rudder deflections needed to track the LOS and is within the constrained limits.

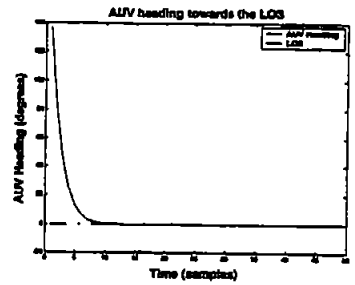


Fig. 6. AUV heading towards and tracking the LOS

Table 1 Simulation parameters for the GA and MPC

Parameter	Value
Prediction Horizon	10
Control Horizon	1
Population Size	50
Number of Generations	50
Mutation Probability	0.005
Crossover Probability	0.7
Q	1
R	0.1

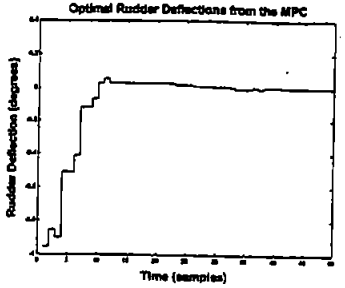


Fig. 7. Optimal rudder deflections generated by the model predictive controller

5. CONCLUSIONS

A new approach to control the yaw angle of an AUV using MPC has been demonstrated. The simple LOS guidance scheme is used to generate the reference heading. The results produced are for stationary targets and are quite encouraging as the actuator constraints are handled in an efficient way. Dealing non-stationary targets using the proposed algorithm is an area of active research. In addition, the technique is currently being employed on MIMO systems, however, this paper deals only with a SISO system.

REFERENCES

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987a). Generalized predictive control. Part 1: The basic algorithm. *Automatica*, vol. 23, no. 2, pp. 137-148.

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987b). Generalized predictive control. Part 2: Extensions and Interpretations. *Automatica*, vol. 23, no. 2, pp. 149-160.

Craven, P. J. (1999). Intelligent control strategies for an autonomous underwater vehicle. *PhD Thesis, University of Plymouth, UK.*

Cutler, C. and B. Ramaker (1979). Dynamic matrix control--a computer control algorithm, AIChE National Meeting, Houston, TX.

Cutler, C. and B. Ramaker (1980). Dynamic matrix control, a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference*, San Francisco, CA.

Duwaish, H. and W. Naeem (2001). Nonlinear model predictive control of Hammerstein and Wiener Models using Genetic Algorithms. In: *Proceedings of the 2001 IEEE International Conference on Control Applications (CCA'01), 5-7 September, Mexico City, Mexico*, pp. 465-469, IEEE.

Millard, N. W., G. Griffiths, G. Finegan, S. D. McPhail, D. T. Meldrum, M. Pebody, J. R. Perrett, P. Stevenson and A. T. Webb (1998). Versatile Autonomous Submersibles- the realising and testing of a practical vehicle. *Journal of the Society for Underwater Technology*, vol. 23, pp. 7-17.

Qin S. J. and T. A. Badgewell (1997). An overview of industrial model predictive control technology. http://www.che.utexas.edu/~qin/ps/cpcv16.ps

Richalet J., J. Testud, A. Rault and J. Papon (1978). Model predictive heuristic control: Applications to industrial processes. *Automatica*, vol. 14, pp. 413-428.

Sait S. M. and Youssef H., (1999). *Iterative computer algorithms with applications in engineering, solving combinatorial optimisation problems*. IEEE Computer Society.

MODEL PREDICTIVE CONTROL OF AN AUTONOMOUS UNDERWATER VEHICLE WITH A FUZZY OBJECTIVE FUNCTION OPTIMIZED USING A GA

W. Naeem, R. Sutton and J. Chudley
{wnaeem, rsutton, jchudley}@plymouth.ac.uk

Marine and Industrial Dynamic Analysis Research Group
Reynolds Building, School of Engineering,
University of Plymouth, Drake Circus,
Plymouth PL4 8AA, UK

Abstract: Recently, unmanned underwater vehicles (UUVs) have emerged as a viable tool for ocean exploration and for military purposes. This is due to the inability of human divers to reach deep sea and the hostile nature of underwater environment. UUVs are of two types, namely, remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs). This paper is concerned with the control of an AUV. A model predictive controller is developed herein where the traditional cost function has been replaced by a fuzzy performance index which represent the goals and constraints of the problem. Since fuzzy logic is basically derived from knowledge of human expertise, it is therefore more intuitive than a conventional cost function. Moreover, the choice of aggregation operator can lead to significant reduction in tuning time which is essential for a quadratic objective function. A genetic algorithm (GA) is used as an optimization tool to evaluate the control inputs by minimization of the performance index represented by fuzzy membership values. The resulting controller is applied to an AUV simulation model obtained from system identification techniques on test trials data. Simulation results are presented that demonstrate the efficacy of the approach.

Keywords: Model predictive control, genetic algorithm, fuzzy objective function, optimization and underwater vehicles

1. INTRODUCTION

"We know what the surface of the moon is better than we know what the surface of the sea floor is."

Designing underwater robots present tremendous challenges to engineers. This is mainly due to the hostile underwater environment and the degrees of freedom of the vehicle movement. The last decade has seen a boost in underwater vehicle development for exploring the rich underwater world containing a huge number of natural resources. As the oceanographer, James Gardner says as quoted by M. Barber, (Barber, 2001) clearly giving a hint that there is still a considerable lack of research work to explore deep oceans. The main hurdle in deep sea exploration is the inability of human divers to reach these places. Underwater vehicles are thought to be a true replacement of deep sea divers for ocean surveying. In addition, they are repeatedly been used in covert missions and for mines clearing operations as the world has recently witnessed the use of the *REMUS* underwater vehicle in the Iraqi conflict (Jordan, 2003).

AUVs are self contained craft that have onboard navigation, guidance and control systems. Thus the range of missions is only limited by the onboard power supply. The navigation system provides information related to the target and vehicle itself, using onboard sensors such as inertial navigation system (INS), compass, pressure transducer etc. This information is fed to the guidance system which by utilising some guidance law generate reference trajectories. The control system is then responsible to keep the vehicle on course as specified by the guidance system. A simple block diagram of the navigation, guidance and control system is depicted in Figure 1.

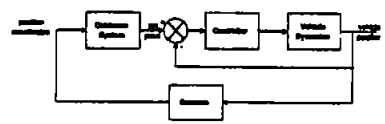


Fig. 1. Navigation, guidance and control of a vehicle

This paper is mainly concerned with the control of an AUV. The control system developed is a genetic algorithm (GA) based model predictive controller (MPC) using fuzzy decision functions. MPC was originated and has long been used in the process industry because of its strong robustness and constraint handling characteristics. As far as the authors are concerned, MPC on AUVs was first simulated by Kwiesielewicz *et al.* (2001) and was compared with a proportional derivative (PD) and an adaptive neuro-fuzzy inference system (ANFIS) tuned autopilot. The results were found to be quite promising. Then Naeem and others simulated a GA tuned MPC on an AUV for heading control (Naeem, 2002) and for subsea cable/pipeline tracking (Naeem *et al.*, 2004). The results demonstrated the robustness of the GA-MPC in the presence of sea currents. The objective function used to evaluate control actions in both cases was a simple quadratic cost function involving the output error, input and the change in input and can be seen in Equation 1.

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \quad (1)$$

subject to

$$u^l \le u(k+i) \le u^u$$
$$\Delta u^l \le \Delta u(k+i) \le \Delta u^u$$

where the superscripts l and u represents the lower and upper bounds respectively. Q is the weight on the prediction error

$$e(k) = \hat{y}(k) - w(k) \quad (2)$$

where $w(k)$ is the reference or the desired setpoint. R and S are weights on the change in the input Δu and magnitude of the input u respectively.

Herein, the control actions are evaluated by using fuzzy membership functions that represent the goals and constraints of the problem similar to a conventional cost function in Equation 1. Fuzzy objective functions in predictive control have been investigated (Sousa and Kaymak, 2001) and the resulting non convex optimization problem was solved using a Branch and Bound (B&B) algorithm. The work presented here is an extension of the previous study.

2. MODEL PREDICTIVE CONTROL

MPC refers to a class of algorithms that compute a sequence of manipulated variable adjustments in order to optimize the future behaviour of a plant. Originally developed to meet the specialised control needs of power plants and petroleum refineries, MPC technology can now be found in a wide variety of application areas including chemicals, food processing, automotive, aerospace and metallurgy (Qin and Badgewell, 2000), to name but a few.

The development of MPC can be traced back to 1978 after the publication of the paper by Richalet *et al.* (1978). They named their algorithm model predictive heuristic control (MPHC) and it was successfully applied to a fluid catalytic cracking unit main fractionator column, a power plant steam generator and a poly-vinyl chloride plant. Then Cutler and Ramaker from the Shell Oil Company in 1979 and 1980 developed their own independent MPC technology referred to as dynamic matrix control (DMC) (Cutler and Ramaker, 1980), and they showed results from a furnace temperature control application to demonstrate improved control quality. However, another form of MPC called generalised predictive control (GPC) (Clarke *et al.*, 1987*a*; Clarke *et al.*, 1987*b*) is employed in this paper. The fundamental difference between all these techniques is the type of model used and the cost function being optimized.

The process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the

output of the process can be used. Further if the process is subject to disturbances, a disturbance or noise model can be added to the process model. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used as defined in Equation 1. The optimal controller output sequence u_{opt} over the prediction horizon is obtained by minimisation of J with respect to u. As a result the future tracking error is minimised. If there is no model mismatch i.e. the model is identical to the process and there are no disturbances and constraints, the process will track the reference trajectory exactly on the sampling instants. The structure of an MPC is shown in Figure 2 and it consists of the following three steps.

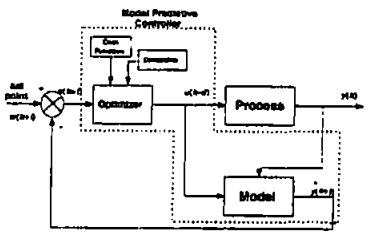


Fig. 2. Structure of a model predictive controller

(1) Explicit use of a model to predict the process output along a future time horizon (Prediction Horizon).
(2) Calculation of a control sequence along a future time horizon (Control Horizon, H_c), to optimize a performance index.
(3) A receding horizon strategy so that at each instant the horizon is moved towards the future, which involves the application of the first control signal of the sequence calculated at each step. The strategy is illustrated as shown in Figure 3.

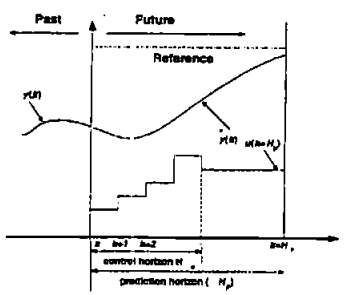


Fig. 3. Predicted output and the corresponding optimum input over a horizon H_p, where $u(k)$ is the optimum input, $\hat{y}(k)$ is the predicted output, and $y(k)$, process output

The selection of MPC to control an AUV is attributed to several factors. Some of them are listed below.

- The concept is equally applicable to single-input, single-output (SISO) as well as multi-input, multi-output systems (MIMO).
- MPC can be applied to linear and nonlinear systems.
- It can handle constraints in a systematic way during the controller design.
- The controller is designed at every sampling instant so disturbances can easily be dealt with.

The performance index is optimized using a GA which is described next.

2.1 Genetic Algorithms

GAs inspired by Darwinian theory, are powerful non-deterministic iterative search heuristics. GAs operate on a population consisting of encoded strings, each string represents a solution. The crossover operator is used on these strings to obtain new solutions, which inherits the good and bad properties of their parent solutions. Each solution has a fitness value, solutions having higher fitness values are most likely to survive for the next generation. The mutation operator is applied to produce new characteristics, which are not present in the parent solutions. The whole procedure is repeated until no further improvement is observed or run time exceeds to some threshold, (Sait and Youssef, 1999). The flowchart of a simple GA is presented in Figure 4.

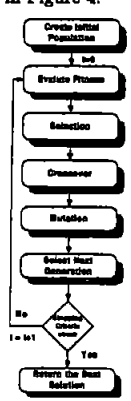


Fig. 4. Flow chart of a simple genetic algorithm

3. FUZZY OBJECTIVE FUNCTION

The fuzzy logic traditionally used as *if-then* rules can be translated to some design specifications using human expertise. These design specifications are represented in terms of an objective function which is more intuitive than conventional cost function. There are several other reasons to choose fuzzy membership values as an objective function to be optimized in a predictive control problem. Some of them are listed below

- Fuzzy objective functions are easy to understand
- Soft and hard constraints can be implemented using the same membership function
- No normalisation is required for the terms in the objective function as the membership function automatically maps the input space to a [0–1] interval
- Easy to tune as the weighting matrices for individual terms are not needed
- The aggregation operator normally requires only a single tuning parameter for all the terms

In this work, membership functions for the output error and input variables are considered. An exponential membership function has been elected for the output error while the input is represented by a trapezoidal membership function as shown in Figures 5 and 6 respectively. The steepness of the exponential plot can be adjusted using the S_e variable and thus is an important tuning parameter. The trapezoidal membership function in Figure 6 automatically implements the soft and hard constraints, where u_{max} represents the maximum allowable input and $(u_{max} - u_{constraint})$ is the input bound which is allowed but not desired.

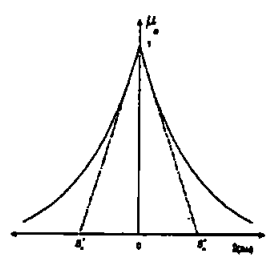


Fig. 5. Output error membership function

3.1 Aggregation Operator

Finally, a decision function is required which allows for interaction amongst different criteria in the objective function. A variety of aggregation operators can be chosen to be used in fuzzy predictive control such as *min* and *product t-norm*. A good account of various aggregation operators and the advantages and disadvantages of their use in

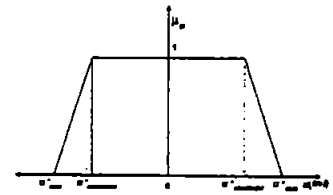


Fig. 6. Trapezoidal membership function for input variable

predictive control have been documented (Sousa and Kaymak, 2001).

Herein, the Yager t-norm has been chosen as the decision function since it uses only one parameter to tune the objective function and hence interact amongst different criteria. Moreover, this operator covers the entire range of t-norms, i.e., it goes from the drastic intersection to the minimum operator (Sousa and Kaymak, 2001). The fuzzy cost function and Yager t-norm are given by Equations 3 and 4 respectively.

$$\bar{\mu}_c = \sum_{i=1}^{H_p} (\bar{\mu}_e(\hat{e}(k+i)))^{w_Y} + \sum_{j=1}^{H_c} (\bar{\mu}_u(u(k+i)))^{w_Y} \tag{3}$$

$$\mu = \max(0, 1 - \bar{\mu}_c^{1/w_Y}) \tag{4}$$

where $\bar{\mu} = 1 - \mu$ and $w_Y > 0$ is the tuning parameter

4. SIMULATION RESULTS

The proposed control algorithm is applied to an AUV simulation model of the *Hammerhead* vehicle being developed as a combined project at the Universities of Plymouth and Cranfield, UK. The vehicle has a torpedo shaped hull approximately 3 metres long and one-third of a metre in diameter. The yaw dynamics of the vehicle has been obtained from test trials at Willen Lake, Milton Keynes, UK and system identification techniques were applied to extract a yaw-rudder channel model given by Equation 5

$$G(q) = \frac{-0.04226q^{-1} + 0.003435q^{-2}}{1 - 1.765q^{-1} + 0.7652q^{-2}} \tag{5}$$

where the data was sampled at a rate of $1Hz$ with the vehicle manoeuvring on the surface at a fixed speed of approximately 2 knots. The following steps describe the operation of a GA as an optimization tool in MPC followed by simulation results for various settings.

(1) Evaluate process outputs using the process model.

(2) Use a GA search to find the optimal control moves which optimize the cost function and satisfy process constraints. This can be accomplished as follows.
   (a) generate a set of random possible control moves.
   (b) find the corresponding process outputs for all possible control moves using the process model.
   (c) evaluate the fitness of each solution using the fuzzy cost function and the process constraints.
   (d) apply the genetic operators (selection, crossover and mutation) to produce new generation of possible solutions. Stochastic universal sampling and single point crossover is used for parents selection and mating respectively.
   (e) repeat until predefined number of generations is reached and thus the optimal control moves are determined.
(3) Apply the optimal control moves generated in step 2 to the process.
(4) Repeat steps 1 to 3 for time step $k+1$.

The hard constraints on the rudder are $\pm 22^{\circ}$ therefore u^{-}_{max} and u^{+}_{max} in Figure 6 are taken as -22° and $+22^{\circ}$ respectively. However, the control is only allowed to move freely within the range $\pm 20^{0}$ to avoid saturation and hence any nonlinear behaviour. Therefore, $u^{\pm}_{constraint}$ in Figure 6 is set equal to $\pm 20^{0}$. Simulations are carried out first for a step change in heading. The vehicle is launched with an arbitrary orientation and is required to follow a specified heading. The parameters S_s and w_Y are chosen as 0.5 and 2 respectively whereas the GA parameters are provided in Table 1 and are selected to minimise the control effort and increase the speed of response. The step response of the closed loop system is depicted in Figure 7 showing that the vehicle is closely following the set point with little overshoot and zero steady state error. The canard demand is also shown in Figure 8 and is within the specified constraints.

Table 1. GA-MPC tuning paramters

Parameters	Step response	Way point following
H_p	10	20
H_c	1	1
Mutation prob.	0.008	0.008
Crossover prob.	0.1	0.1
No. of generations	1	3
Population size	100	250
Insertion rate	0.5	0.1

Next, the control law is simulated for way point following where the intent is to track all the specified way points despite the presence of disturbances. A sea current disturbance is assumed to be acting on the vehicle in the positive y-direction

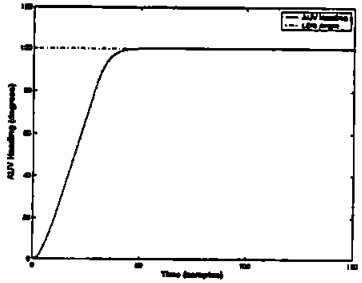


Fig. 7. Step change in heading response of the proposed controller.

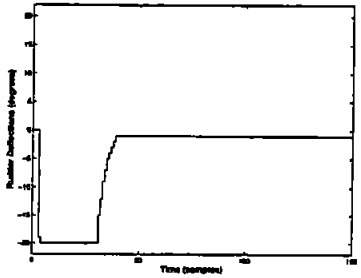


Fig. 8. Optimal rudder deflections generated by the proposed controller.

and has a magnitude of $0.5m/s$. The initial AUV coordinates in the two-dimensional frame of reference is $(0, 10)$ while the way points are chosen to be at $(100, 50)$, $(300, 50)$, $(500, 150)$ and $(500, 300)$. The next way point is selected when the vehicle enters a circle of acceptance around the way point of radius $10m$. As will be shown, the way point following is equivalent to tracking line of sight (LOS) angle between any two given way points. The GA parameters for this case are also given in Table 1 while S_s and w_Y are selected as 1.5 and 1.8 respectively. The resulting closed loop performance is illustrated in Figure 9 showing the affects of sea currents on vehicle's trajectory. The disturbance is striving to knock the AUV off the track, however, the controller is quite robust to follow all the way points. The AUV trajectory through the way points without any sea currents is also illustrated in Figure 9 which is closely following the ideal path. The control effort depicted in Figure 10 shows vigorous rudder movements in response to the change in vehicle's heading due to sea currents, however, it never violates the imposed constraints. The heading angle or LOS angle between the actual AUV position and way points shown in Figure 11 is varying continuously because of the addition of disturbances.

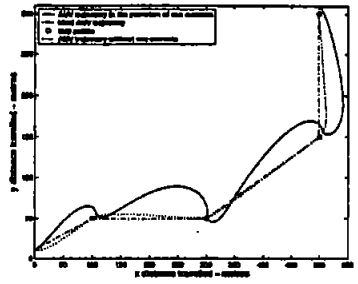


Fig. 9. AUV and target position co-ordinates with sea current disturbance in the positive y-direction.

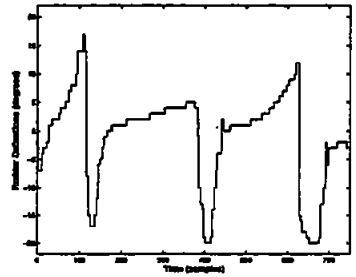


Fig. 10. Rudder deflections generated by the controller needed to track the way points with sea current disturbance in the positive y-direction.

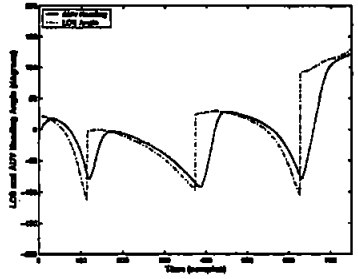


Fig. 11. Desired heading angles to the way points showing the affects of sea current disturbance

CONCLUDING REMARKS

The traditional fuzzy logic based on if-then rules has been used for decision making in an MPC framework to control an AUV. A GA is used to optimize the resulting fuzzy objective function. The advantages of using the proposed scheme are presented. The GA optimized fuzzy MPC have been applied to an AUV simulation dynamic model based on the *Hammerhead* vehicle being developed jointly at the Universities of Plymouth and Cranfield, UK and will eventually be tested on the real vehicle.

REFERENCES

Barber, M. (2001). *Hydrographic crew surveys underwater quake damage*. World Wide Web. `http://seattlepi.nwsource.com`.

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987*a*). Generalised predictive control. Part 1: The basic algorithm. *Automatica* **23**(2), 137–148.

Clarke, D. W., C. Mohtadi and P. S. Tuff (1987*b*). Generalised predictive control. Part 2: Extensions and interpretations. *Automatica* **23**(2), 149–160.

Cutler, C. R. and B. L. Ramaker (1980). Dynamic matrix control - a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference, Paper WP5-B*. San Fransisco, CA, USA.

Jordan, K. (2003). *Remus AUV Plays Key Role in Iraq War*. World Wide Web. `http://www.diveweb.com`.

Kwiesielewicz, M., W. Piotrowski and R. Sutton (2001). Predictive versus fuzzy control of autonomous underwater vehicle. In: *Proceedings IEEE International Conference on Methods and Models in Automation and Robotics*. IEEE. Miedzyzdroje, Poland. pp. 609–612.

Naeem, W. (2002). Model predictive control of an autonomous underwater vehicle. In: *Proceedings UKACC Control 2002*. UKACC. pp. 19–23.

Naeem, W., R. Sutton and S. M. Ahmad (2004). Pure pursuit guidance and model predictive control of an autonomous underwater vehicle for cable/pipeline tracking. *IMarEST Journal of Marine Science and Environment* C(1), 25–35.

Qin, S. J. and T. A. Badgewell (2000). An overview of nonlinear model predictive control applications. In: *Nonlinear Model Predictive Control*. Birkhäuser. Switzerland. pp. 369–392.

Richalet, J., A. Rault, J. L. Testud and J. Papon (1978). Model predictive heuristic control: Applications to industial processes. *Automatica* **14**, 413–428.

Sait, S. M. and H. Youssef (1999). *Iterative Computer Algorithms with Applications in Engineering, Solving Combinatorial Optimization Problems*. IEEE Computer Society.

Sousa, J. M. and U. Kaymak (2001). Model predictive control using fuzzy decision functions. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics* **31**(1), 54–65.

ADAPTIVE AUV CONTROL FOR OPTIMIZED SWATHE LASER STRIPE IMAGING

F R Dalgleish*, W Naeem†, S Tetlow*, R L Allwood*, R Sutton†

**Offshore Technology Centre, Cranfield University, Beds, UK*

†School of Engineering, The University of Plymouth, Plymouth, Devon, UK

Abstract: As part of a three year collaborative research project funded by EPSRC involving both Cranfield University and the University of Plymouth, a Deep Mobile Target (DMT5) torpedo-shaped underwater vehicle has been developed into an AUV called *Hammerhead*. In addition to several low cost navigation sensors, the AUV has been installed with a laser-assisted imaging and navigation sensor. This sensor is based on the laser stripe illumination (LSI) methodology previously developed at Cranfield University (Tetlow and Allwood, 1995), providing enhanced viewing of the seabed below the vehicle. However, it also provides real time altitude and seabed-relative translational velocity data to the navigation system during the mission (Dalgleish et al., 2003), as well as gather images to produce a post mission enhanced optical waterfall image of a surveyed area (Dalgleish et al., 2004).

For a particular water type and clarity, the optimal stand-off distance (altitude and hence swathe) between the LSI and the seabed will vary. The optimisation problem is essentially the pursuit of a balance between image survey efficiency and image quality. This paper describes the development of an automatic process for monitoring the LSI image quality to determine the optimum altitude at which to fly, and henceforth instruct the AUV control system to do so. In this way, swathe may be optimised for maximum seabed coverage and more efficient use of AUV time. The paper also presents simulation results from a genetic algorithm (GA) based model predictive control (MPC) methodology used to track the required altitude provided by this process.

Keywords: navigation, autonomous, real-time, image sensors, model-based control, predictive control, image processing

1. INTRODUCTION

1.1 Background

Although underwater vision systems only cover a small area when compared with wide swathe acoustic systems, the high resolution and ease of interpretation offered by optical techniques may be necessary for certain seabed survey or identification tasks.

However, the performance of these underwater vision systems is affected by the scattering and absorption properties of the water through which they are viewing. Other than in shallow water during daytime when ambient lighting may be sufficient, light has to be provided artificially. The conventional solution, which usually consists of a wide-angle lamp as a source, is limited by both:

(i) absorption of the light as it propagates through water to the target (and back again).
(ii) near-field backscatter of the light particularly when the source and receiver are in close proximity.

In the case where the intensity of scattered light is such that it veils the image, the system is said to be 'contrast limited'. An increase in the separation between the camera and the lamp can reduce the near-field backscattered light, but then the system may become limited by absorption where the returning signal is too weak to be detected. In this case the system is said to be 'power limited'. Increasing the power of the lamp does not significantly improve the situation due to the exponential decay of light per unit distance. Furthermore, the spreading effect of the light beam and the finite aperture of a camera lens further reduce the incoming power to the receiver.

Choosing the source wavelength to match the optimal blue-green transmission window of water, has driven research efforts towards monochromatic light sources, in particular green lasers, where several systems have been designed and tested. These systems fall into three main categories:

(i) range gated
(ii) synchronous scanning
(iii) non-synchronous scanning

Range gating exploits the 'time of flight' of an illuminating light pulse in seawater and requires precise gating of both the light source and the receiver. (Fournier et al., 1994) (Swartz at al., 1993)

Synchronous scanning or Laser Line Scan (LLS) methods exploit the highly collimated nature of a laser beam by using a narrow field of view (FOV) receiver to track the spot at high speed, thus reducing backscattered and forward scattered light. (Strand et al., 1996) (Coles 1997)

A less expensive and sophisticated alternative is to optically or mechanically form a fan of light as a source and use a standard receiver such as a low light charged coupled device (CCD) camera with a wide angle lens. Image processing techniques can be used to partially remove the forward and back scattered light. These methods are known as non-synchronous scanning systems or laser stripe illumination systems (LSI), of which a couple of systems have been developed (Tetlow & Allwood, 1995) (Langebrake et al., 2000). In addition, more sophisticated pulsed laser variations exist (Moore et al., 2000).

From computer simulations (Jaffe and Dunn, 1988) these non-synchronous scanning systems become limited between 5 and 7 attenuation lengths, compared with 2 to 4 for a conventionally illuminated system.

The Cranfield LSI system has been found to significantly improve image contrast over conventional imaging (Tetlow, 1993). Furthermore, the images produced are approximately optically flat, meaning they exhibit even illumination across the horizontal FOV. Also, the greater depth of view allows mid-water objects such as fish or objects at various distances to be seen, and the structured nature of the illumination allows seabed bathymetry to be derived from the image. Likewise, from the intensity of the returning light it is possible to produce simultaneous reflectivity profiles of the seabed.

Due to the fact that the Cranfield LSI system uses the entire field of view (FOV) of a standard 2-D low light CCD camera, there is much more information being gathered than just the stripe (as long as the area outwith the stripe is being illuminated either by ambient light or an additional lamp, and these areas of the image are not contrast limited). As the vehicle passes over the seabed it is possible to track the movement of certain regions (which contain features or patterns of texture) through the FOV between successive images. Knowing the distance to the seabed (altitude) from the laser triangulation process, it is possible to determine the displacement of the vehicle over the ground, and with accurate knowledge of the time between images, 3-D terrain-relative velocity profiles can be generated. (Dalgleish et al., 2003)

One of the distinct differences of the LSI system over Laser Line Scan (LLS) system is that due to the fact that the receiver is a standard 2-D CCD camera with a wide field of view, the system has a range of altitudes above the seabed over which it is able to operate (3 to 18 metres for a recent configuration), rather than having a tightly constrained laser triangulation geometry consistent with a 1-D CCD camera or narrow FOV photo-multiplier tube (PMT) that are utilised by LLS systems, requiring complex electro-optical aperture assemblies which are shaved off onboard altimeters. Indeed, some of these systems even require the detector and receiver geometry to be manually adjusted prior to the mission to be able to capture images. This adjustment is based on knowledge of the water type and from previous experience.

Being a simple robust system, with a minimum of moving parts, it was felt that LSI would be an ideal imaging system to install on an AUV to be used in medium turbid to clear water.

1.2 EPSRC Funded Hammerhead AUV Programme

As part of a three year collaborative research project funded by EPSRC involving both Cranfield University and the University of Plymouth, a Deep Mobile Target (DMT5) torpedo-shaped underwater vehicle has been developed into an AUV with an integrated LSI system. The AUV is called *Hammerhead*.

The navigation system being developed at the University of Plymouth is based on a multi-sensor data fusion (MSDF) algorithm using a variety of low cost sensors. During an actual surveying task this subsystem will be enhanced by data from the vision system, with the purpose of aiding navigation and optimising the imaging parameters. Once the navigation data has been suitably processed it is fed to the guidance and control system for the appropriate action.

A summary of the AUV data and image acquisition system and integrated imaging system is given in table 1. The motivation behind using a small form factor PC-based system was driven by the perceived ultimate use of the AUV; that being an academic test-bed vehicle where it is appropriate that applications developed on PC-based hardware can be tested directly and easily on the AUV.

The adjustment to imaging parameters and switching the laser and lamp on and off is either performed manually (on the surface) via a wireless Ethernet link

or automatically from the host computer based on an image quality control computer vision application or a trigger from the onboard sonar altimeter. The entire system is powered from both ±12 volt and 12 volt supplies provided by lead acid batteries.

Table 1. Summary of Hammerhead AUV data and image acquisition system with integrated LSI

Laser	100mW (ow) frequency doubled diode pumped Nd:YAG (532nm)
Light	50W dichroic halogen (12V)
Camera	Monochrome 1/3" low light CCD Sensistivity 0.0001 Lux (AGC on) Automatic control of AGC (on/off)
Scanner	80Hz resonant scanner sinusoidal driving signal
Lens	Automatic control of iris, focus and zoom
Compass	TCM2 tilt-compensated electronic compass
Frame Grabber	National Instruments PCI-1409 Image size (pixels) is 576x768. Image depth is 8 or 10 bit
Data Acquisition	National Instruments PCI-6024E
Computer	MicroATX motherboard AMD 2800+ processor 1024MB RAM 120GB Hard Drive
Software	Windows™ 2000 Professional LabVIEW 6.1 Full Development System IMAQ Vision Development Module 6.0.1

1.3 Optimised Swathe LSI

For a particular water type and clarity, the optimal range between the LSI and the seabed will vary. As the range (or altitude) increases, the swathe (or width of a single line of survey coverage) also increases. The optimisation problem is essentially the pursuit of a balance between image survey efficiency and image quality. By adopting a higher survey altitude, LSI allows a larger swathe to be surveyed. This reduces the time needed to survey a particular area. However, the quality of the images produced may not be optimal. It is clear that there is a balance to be met in order to optimise the system.

It can be seen from figure 1 that the FOV is proportional to the altitude (Z).

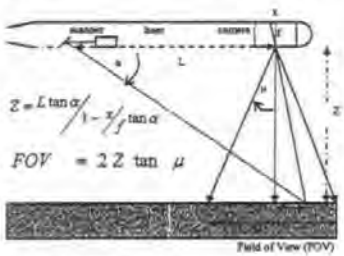


Fig. 1. Geometry for triangulation ranging and swathe width determination.

Therefore, if it is possible to increase the imaging range of operation by one metre or 20%, for example, then this will approximately reduce the survey time by the same amount. For the Hammerhead AUV, which cruises at 2.5 knots, the theoretical time taken to 'lawnmower' survey a square nautical mile at a nominal altitude of 5 metres with 10% survey line overlap and a 30 second turn around time is approximately 50 hours. With the swathe optimisation applied the time saved is 10 hours.

Now consider the case when the LSI system is too close to the seabed and the stripe is too bright leading to 'blooming' (when a pixel overflows with electrons and leaks into adjacent pixels) in the CCD camera adversely affecting image quality. Although it is possible to automatically close the iris, in this case, it is more effective, and indeed efficient to increase the altitude[1].

Conversely, when the LSI system is too far away from the seabed, the situation can either become power limited or contrast limited, depending on the turbidity of the water. For the purpose of this paper, where clearer water is being considered, the power limited scenario is investigated only. The role of the vision algorithm is to determine the optimality of the images (and resulting composite images) based on the mean intensity value of the extracted laser stripes, and to instruct the AUV control system to alter its altitude appropriately.

1.4 Altitude controller

A genetic algorithm (GA) based model predictive control (MPC) methodology has been developed to track the required altitude provided by the imaging sensor. The MPC technique has previously been successfully implemented on the Hammerhead vehicle in the horizontal plane (Naeem et al., 2004). Herein, an MPC altitude (depth) controller has been developed to track the guidance commands from the vision system. Due to lack of space, the reader is referred to Naeem et al. (2004) for a comprehensive review on the design of a robust GA based controller.

A Hammerhead depth dynamic model was extracted using system identification techniques on actual vehicle trials data. In this case, since the requirement is to maintain a certain altitude (not depth), the depth information can be extracted from the altitude data. In practice, the pressure sensor on board the AUV provides the depth of the vehicle below the sea surface whereas the imaging sensor provides the altitude information above the sea bed. These two quantities are added to obtain the depth of the sea bed. The desired altitude is then subtracted from the result to find the desired depth.

1.5 Overview

This paper describes a heuristic intensity-based technique to automatically determine if the image parameters are optimal, along with the AUV control demand required to remedy the situation. Simulation results for the MPC GA altitude controller are then presented. Finally, the future work necessary to properly develop and test the system is presented, together with real-time AUV control uses of the techniques.

[1] Interestingly, closing the iris increases the depth of focus thus bringing the backscatter field into focus, with the effect of further degrading the image.

2. LASER STRIPE CHARACTERISTICS

The LSI system onboard the Hammerhead AUV is used for both imaging and navigation purposes. For 2-D composite waterfall imaging, such as the ones shown in figure 2, it is necessary to be able to extract the stripe by intensity threshold techniques, whilst retaining the maximum amount of shape and intensity detail within the stripe as possible. For navigation purposes (altitude determination), shown in figure 1, it is the accuracy at which the stripe itself can be extracted from the image that is important.

In order to satisfy both requirements, the emphasis is on accurately extracting the stripe whilst preserving the intensity variation within the stripe, thus not losing any visual information. From a set of trials at the Deep Wave Basin at IFREMER in Brest (Dalgleish et al, 2003), it was possible to take some images at various altitudes. Figure 2 shows a composite waterfall image which was constructed by stitching the extracted stripe region together from a sequence of 600 consecutive images taken in filtered seawater at an altitude of 6 metres and velocity of 0.5 ms^{-1}. With a horizontal angle of view (in water) of over 40 degrees, the swathe was 4.5 metres wide. This composite image contains a wide intensity variation.

Fig. 2. a waterfall image with sufficient intensity information.

As the facility at Brest contained filtered seawater, the level of scattering was low. Examples of these raw images which were taken at night with the camera automatic gain control (AGC) set to 'off', are shown in figure 3.

It can be seen that as the altitude increases, the intensity of the stripe reduces (mainly due to absorption effects). The image at 4 metres exhibits slight 'blooming', whilst the image taken at 16 metres has an intensity level lower than 50.[2]

Fig. 3. Raw images taken at different altitudes with zero ambient lighting. (clockwise from top left: 4m, 8m, 12m, 16m, 14m, 10m)

Figure 4 shows the variation in mean intensity of the stripe at different altitudes.

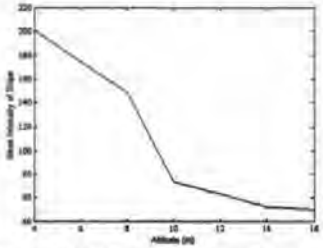


Fig. 4. Mean intensity of laser stripe at different altitudes

The mean intensity of the stripe and the corresponding position (to determine the altitude) is calculate by detecting the peak value of the returning signal from the seabed to sub-pixel accuracy using an algorithm that fits a quadratic polynomial to the stripe region of each vertical column of the image array. The number of data points used in the fit is specified by width. For each peak, the quadratic fit is tested against the threshold level. Peaks with heights lower than the threshold level are ignored. Once the peak intensity value has been established from each column, the mean can be calculated.

With the exception of the image taken at 4 metres, the thickness of the stripe in pixels is approximately constant for the remaining images. This is due to laser beam divergence being cancelled out by the effect of increasing the range and therefore the width of the target.

[2] Throughout this paper, as the images used were 8 bit, the maximum intensity is a decimal value of 255.

3. AUTOMATICALLY DETERMINING OPTIMUM STRIPE CHARACTERISTICS

It can be seen from the clear water images that very little scattering is present. This allows the extraction of the stripe by intensity threshold techniques, even down to low stripe intensity values. However, at these levels of intensity the imaging capabilities are not optimal, and it is likely that important visual information may still be lost. In order to avoid this it is necessary for the system to determine and instruct the AUV to fly at an altitude at which the intensity of the stripe is optimal, both in terms of ease of extraction and of imaging quality.

When manually adjusting the threshold level in order to robustly extract the stripe from an image which contains a distinct stripe, it is usual to use a threshold value of approximately 100 in order to reject extraneous light. Unless highly reflective targets are present, the maximum intensity level is unlikely to saturate at 255, perhaps peaking at 250. Therefore the range of greyscale levels for the stripe is approximately 150. In the optimal case and assuming average target reflectivity along the length of the illuminated region, the mean value of the extracted stripe is likely to be about 175, which is a useful and readily available metric for this purpose. Furthermore, it can be seen in figure 4 that a mean intensity value of 175 corresponds to an altitude of 6 metres (N.B. the acceptable image in figure 2 was taken at 6 metres).

It could be argued that prediction of the stripe thickness is a useful tool to aid in the stripe extraction process. However, it has previously been determined that beam divergence is largely affected by water turbidity and target reflectivity, rendering it unreliable for this purpose (Spours, 2000) (MacAdam-Sproat, 2004).

The use of the mean intensity along the extracted laser stripe is the real essence of this approach.

4. SIMULATION RESULTS

It is assumed that the vehicle was launched from the surface and is required to follow a certain altitude as commanded by the vision system to maintain the image quality[3]. The trajectory or guidance commands obtained by the imaging sensor and the Hammerhead response are depicted in Figure 5.

Clearly, the trajectory generated is not entirely smooth. However, the controller is capable of following it, ignoring the high frequency contamination which is a measure of the robustness of the controller to external disturbances. The control effort (hydroplane deflections) produced by the controller is illustrated in Figure 6 which never violates the actuator constraints of $\pm 25^{0}$. The change in deflection angle is also evident from the figure as the vehicle manoeuvres with respect to the change in reference depth.

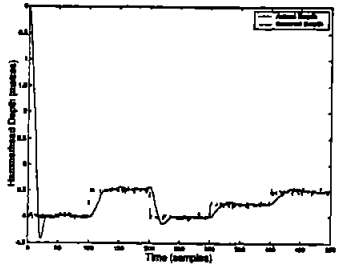


Fig. 5. The trajectory or guidance commands obtained by the imaging sensor and the AUV response.

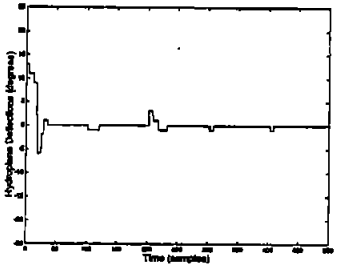


Fig. 6. Hydroplane deflections produced by the controller.

5. CONCLUSIONS AND FUTURE WORK

The discussion in this paper is based on images obtained during a set of trials where the water turbidity was low (the total attenuation coefficient being estimated at $0.2m^{-1}$), typically oceanic. Therefore only the power limited scenario is considered. However, the author and others are currently performing seawater acceptance trials for the Hammerhead AUV, and it is herein intended to further refine the swathe optimisation algorithms to accommodate the likely contrast limited scenario which can result from the higher level of scattering due to these coastal conditions. This is in anticipation of real-time field testing of the swathe optimisation controller, the promising initial results from which are described in this paper.

In turbid estuarine water where the total attenuation coefficient can be perhaps $0.6m^{-1}$ and the situation become contrast limited, the utility of the LSI system is reduced significantly. Nevertheless, work exists in which the task of extracting the stripe in very turbid water is addressed. (Tetlow, 1993) (MacAdam-Sproat, 2004). However, the methods used are not deemed suitable for real-time image processing, such as that required when deploying the LSI system from an AUV.

Having extracted the stripe from a raw image, the most important information contained in the stripe is the amount of intensity detail. In average turbidity water it has been determined that approximately only the top quarter of greyscales are used in the construction of the composite images. This is due to the fact that it is necessary to threshold above a certain value to exclude the noise from around the stripe region, and that the maximum intensity along the stripe in an optimum image is usually slightly less than the highest white value. The work in this paper used 8 bit images. Therefore this means that only about 60 different levels of greyscale are represented in the waterfall images. Although it is possible to improve the dynamic range of the waterfall images by stretching the contrast (Nevis, 1999), a more suitable solution is found in using a higher bit depth of digitisation. Indeed, in the forthcoming trials the author has made provision for 10 bit images.

In the near future, a set of confined water trials have the aim of establishing the turbidity limits of operation of the system in terms of extracting the stripe for accurate altitude determination. The resolving capability of the composite imaging algorithms will also be evaluated as the turbidity increases. In order to validate a system model, the total attenuation coefficient will be measured during these experiments. When performing open water trials the altitude of the AUV will also be measured with a sonar altimeter, which will be compared to the LSI system. In very turbid water this will also be used to prevent the AUV attaining too low an altitude, and risking collision with the seabed. The sonar altimeter may also be used in increasing the efficiency in extracting the laser stripe in these turbid conditions, by predicting its location in the image.

The techniques described here are a step towards a real time implementation of waterfall imaging. Indeed, when capable of performing extended seabed imaging missions, it is intended to produce composite waterfall maps in real time and develop strategies to overcome the drift in vision-based navigation by revisiting the previously imaged areas within the overlap of a particular type of search pattern.

---

[1] The authors would like to acknowledge the help received from IFREMER. The work was funded by the EU under the Access to Research Infrastructure action of the Improving Human Potential Programme.

[2] The authors would also like to thank EPSRC for funding this research project.

[3] The onboard sonar altimeter is used to guide the vehicle until a visual return from the seabed is detected

REFERENCES

Coles, B., (1997). Laser line scan systems as environmental survey tools. *Ocean News Technology*, 3, pp 22-24

Dalgleish, F.R., S. Tetlow and R.L. Allwood (2003). Experiments in laser assisted vision sensing. Submitted to *Control Engineering Practice*.

Dalgleish, F.R., S. Tetlow and R.L. Allwood (2004). A laser-assisted sensor for AUV navigation. *Proc. ISOPE 2004*, 23rd – 28th May, 2004, Toulon, France.

Fournier, G.R., D. Bonnier, J. Luc Forand, and P. W. Pace (1993) Range-gated underwater laser imaging system. *Opt. Eng.* 32, pp 2185-2190.

Jaffe, J. and Dunn, C. (1988). A Model-Based Comparison of Underwater Imaging Systems. *SPIE*, vol. 925, *Ocean Optics IX*, pp. 344-350.

Langebrake, L. C. S. A. Samson, E. A. Kaltenbacher, E. T. Steimle, J. T. Patten, C. E. Lembke R. H. Byrne, K. Carder, T. Hopkins. (2000). Sensor Development: Progress towards systems for AUVs/UUVs. *MTS/IEEE Oceans 2000*. 12th -14th Sept, 2000. Providence, R.I.

MacAdam-Sproat, G.O. (2004). The development of machine algorithms for the Cranfield underwater laser stripe imaging system. *PhD Thesis, Cranfield University, UK.*

Moore, K.D., J. S. Jaffe and B. L. Ochoa (2000) Development of a new underwater bathymetric laser imaging system: L-Bath, *Journal of Atmospheric and Oceanic Tech.* Vol. 17, No. 8, pg. 1106-1117.

Naeem, W., R. Sutton, J. Chudley, F. R. Dalgleish and S. Tetlow (2004). A GA based model predictive control autopilot design and its implementation in an autonomous underwater vehicle. *Accepted in the IMeChE Transactions Part M, Journal of Engineering for the Maritime Environment.*

Nevis, A.J. (1999). Adaptive background equalisation and image processing applications for Laser Line Scan data. *SPIE, 3710, pp 1260-1271.*

Spours, J. (2000). The use of a structured laser light system to ascertain three dimensional measurements of underwater work sites. *PhD Thesis, Cranfield University, UK.*

Strand, M.P., (1995) Underwater electro-optical systems for mine identification. *Proc. Autonomous Vehicles in Mine Countermeasures.* Monterey, CA., Naval Postgraduate School, 6-238-6-247.

Swartz, B., R. Morton, and S. Moran, (1993). Diver and ROV deployable laser range gated underwater imaging system. Proc. MTS *Underwater Intervention '93.*

Tetlow, S.W. (1993). Use of laser light stripes to reduce backscatter in an underwater viewing system. *PhD Thesis, Cranfield University, UK.*

Tetlow, S. and R. L. Allwood (1995). Development and applications of a novel underwater laser illumination system *Underwater Technology* Vol 21, No 2, Autumn 1995, pp 13-20

OBSERVER KALMAN FILTER IDENTIFICATION OF AN AUTONOMOUS UNDERWATER VEHICLE

A. Tiano [*], R. Sutton [+], A. Lozowicki [§] and W. Naeem [+]

[*] *University of Pavia, Department of Information and Systems, Via Ferrata 1, 27100 Pavia, Italy*
[+] *University of Plymouth, Dept. of Mechanical and Marine Engineering, PL4 8AA, Plymouth, UK*
[§] *Technical University of Szczecin, ul. Piastow 17, PL-70313 Szczecin, Poland*

Abstract: This paper discusses the identification of linear multivariable models of an autonomous underwater vehicle. The OKID (Observer Kalman Identification) method is applied with the main objectives of verifying couplings between different motions and to evaluate its applicability to the design an LQG control system. The method is tested on the basis of experimental data obtained from experiments on an autonomous underwater vehicle. Some preliminary identification results are presented and commented.

Keywords: system identification, parameter estimation, numerical methods, underwater vehicles.

1. INTRODUCTION

An increasing interest has been devoted in the recent years to the experimental determination of the dynamic behaviour of underwater vehicles by means of system identification methods. In many situations it is important that the mathematical models obtained by means of system identification methods can be directly used for control system design purposes.
This implies that linear models that take into account couplings between different motions should be determined. In fact, in such case linear control algorithms can be easily implemented and the effect of neglected dynamics can be minimized. An identification method that can cope with these requirements is OKID (Juang, 1994), that has proven to be also numerically very efficient and robust with respect to measurement noise. In this paper OKID will be applied to the depth dynamics of the autonomous vehicle Hammerhead (Naeem et Al., 2003) by using real sea trials data.

The paper is organized as follows. After the introduction, the underwater vehicle will be shortly described in section 2, where also the experiments conducted on the vehicle will be discussed. Section 3 is dedicated to the description of the identification method and section 4 is dedicated to the illustration of the preliminary identification results, while some concluding remarks are expressed in section 5.

2. UNDERWATER VEHICLE

The vehicle considered for identification , called *Hammerhead* (Naeem et Al., 2003), has a torpedo shaped body, is about 3.5 meters long and has a diameter of approximately 0.3 meters. The control surfaces are the two rear rudders for steering and two front hydroplanes for diving. These control surfaces are controlled by two separate onboard stepper motors and the signal to the stepper motors is sent through an umbilical cable attached to the rear end of the vehicle. The onboard sensors include an inertial

navigation system (INS), a TCM2 compass, pressure sensor, global positioning system (GPS), and a shaft speed encoder. The data logged using the above mentioned sensors is summarized below:

- INS (heading, pitch, roll, linear and angular velocities)
- TCM2 compass (heading, pitch and roll angles)
- Pressure sensor (depth of the vehicle)
- GPS (cordinates of the vehicle on the sea surface, forward spewed)
- Shaft speed encoder (vehicle speed)

The vehicle is connected through a rear umbilical cable to a control computer used to transmit various input/output signals. As a first attempt, the rudder/heading angles data were used to identify a longitudinal model, while the hydroplane angle/depth data were used to identify a lateral model (Naeem et Al. 2003).However, if cross couplings between yaw and roll and between depth and pitch result not negligible, coupled multivariable models hould be used.

During the experiments different multistep inputs were applied to the control surfaces and the output variables were recorded with a sampling frequency of 8 Hz. Some examples are shown in Fig. 1 and Fig.2.

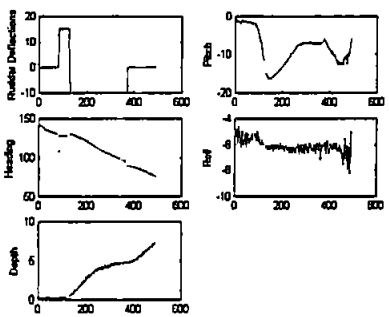


Fig 1. Example of input/output data after an identification experiment of depth dynamics.

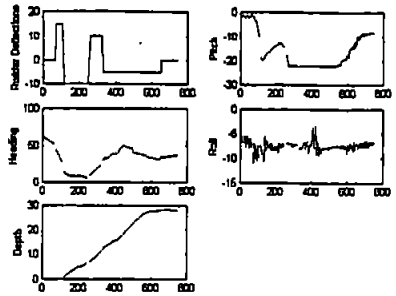


Fig 2. Example of input/output data after an identification experiment of depth dynamics.

The above figures are related to experiments aiming at the identification of the lateral dynamics, for determining the depth and pitch response to the sternplane deflections. As it can be observed , owing to the limitations of the software used for data acquisition, some missing data occurred. This inconvenience was solved by using an interpolation algorithm (Kybic et Al., 2001) deduced by a variational approach.

As it can be noticed, there exist indeed appreciable couplings between such motions. However, since this paper aims at the preliminary objective of verifying the suitability of OKID identification method for AUV dynamics, only depth response will be identified.

3. IDENTIFICATION METHOD

A linear time-invariant discrete time MIMO state space model describing expressing the relation between the input vector $u(k) \in R^m$, the output vector $y(k) \in R^p$ and the state vector $x(k) \in R^n$ can be expressed as :

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + v(k) \\ y(k) &= Cx(k) + Du(k) + e(k) \end{aligned} \quad (1)$$

Such equation includes also the effects of a disturbance vector $v(k)$ of measurement noise vector $e(k)$.

The Observer Kalman Filter Identification, (OKID) method (Juang,1994), attempts to identify, on the basis of input/output data, the smallest state space realization that is compatible with a given accuracy. For this purpose an input/output description of the above system is assumed of the form:

$$y(k+1) + \sum_{i=1}^{n} M_i^{(2)} y(k-i) = \sum_{j=1}^{n} M_j^{(1)} u(k-j) + \varepsilon(k) \quad (2)$$

where $M_i^{(2)} \in R^{p \times p}$, $i = 1,..n$ and $M_j^{(1)} \in R^{p \times m}$, $j = 1,..n$ are the Markov matrices for a model with order n and $\varepsilon(k)$ is the filter residual vector.

Equation (2) is the MIMO linear model that can be used for estimating the joint Markov parameters

$$M_k = \left[M_k^{(1)} \quad -M_k^{(2)} \right] \in R^{p \times (p+m)}, \; k = 1,..n \quad (3)$$

by a recursive Least Squares algorithm.

Once Markov parameters have been determined, the Hankel matrix can be constructed:

$$H(k-1) = \begin{bmatrix} M_k & M_{k+1} & .... & M_{k+s-1} \\ M_{k+1} & M_{k+2} & .... & M_{k+s} \\ . & . & .... & . \\ M_{k+s-1} & M_{k+s} & .... & M_{k+2s-2} \end{bmatrix} \quad (4)$$

By using the ERA/DC (EigenSystem Realization Algorithm with Data Correlation) (Juang et Al., 1988), it is possible to obtain a block correlation Hankel matrix composed of correlation matrices :

$$R(k) = H(k)H^T(0) \quad (5)$$

Such correlation matrices, are easily derived by Markov parameters and allow to reduce noise effects on the identification procedure.
Finally a block correlation Hankel matrix $W(k)$ associated to $R(k)$ matrices is determined, from which, by applying an SVD (Singular Value Decomposition), the unknown system order n is determined, since it is coincident with the number of positive singular values. If $W(0) = U_n S_n V_n^T$ is such SVD decomposition, it follows that a possible set of realization matrices of the identified system is:

$$\begin{aligned} A &= S_n^{-1/2} U_n^T W(1) V_n S_n^{-1/2} \\ B &= S_n^{1/2} V_n^T E_m \\ C &= E_p^T U_n S_n^{1/2} \end{aligned} \quad (6)$$

where matrices E_m and E_p are block selection matrices constituted by block of identity and zero matrices.

It can also be shown that it is possible to implement an observer applied to the deterministic version of linear state space equation (1), by considering the extended system :

$$\begin{aligned} x(k+1) &= \hat{A}x(k) + \hat{B}v(k) \\ y(k) &= Cx(k) + \hat{D}v(k) \end{aligned} \quad (7)$$

with the extended input $v(k) = \begin{bmatrix} u(k) \\ y(k) \end{bmatrix}$ and modified matrices :

$$\hat{A} = A + GC, \; \hat{B} = B + GD - G, \hat{D} = [D \quad 0] \quad (7)$$

where all the eigenvalues of the modified system matrix $\hat{A}$, under an observability hypothesis, can be placed in the origin. In this case, it can be recognized that the gain matrix G plays a role analogous to Kalman filter in state estimation.
By using equation (6) as a state observer, it is also possible to implement an LQG controller (Juang,1994).

4. IDENTIFICATION RESULTS

A number of depth response experimental files have been used for identification by using the OKID method. In Fig.3 there is shown an example of sternplane input time history, while in Fig.4 the singular value magnitudes are plotted. As it can be noticed there is an appreciable drop in magnitude of the singular values between 7 and 8, indicating a system order n of 7 . It is worth noting that evidently an overestimation of the system order has been achieved. This fact derives from having neglected the coupling between depth and pitch motions. Furthermore, the depth measurement error seems to be relatively high, thus contributing to the increase of the estimated order.

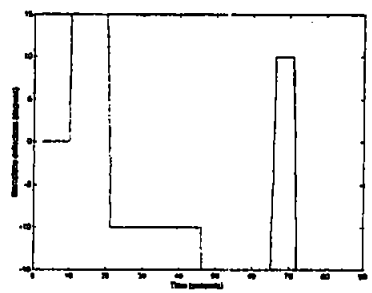


Fig. 3. Sternplane multi step input

It should be noticed, however, that as far as identification aims at determining a model for control purposes, overestimation of the model order is not a real drawback, since in this case identification is mainly used for predictive purposes. In Fig. 5 a comparison of the measured depth with the

corresponding one predicted by the identified model is shown. It can be appreciated the robustness of OKID algorithm, even in the presence of a relatively high measurement noise on the depth channel.

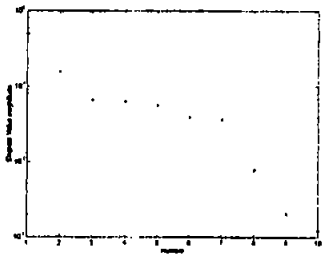


Fig. 4. SVD magnitude vs model order.

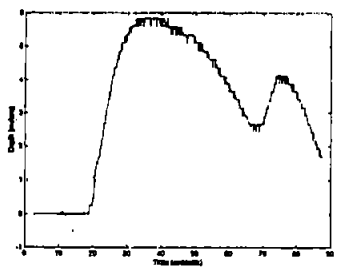


Fig. 5. Comparison between measured depth and predicted depth.

Analogous results have been obtained after the identification of different experiments.

5. CONCLUDING REMARKS

In this paper the OKID method has been applied to the identification of the depth dynamics of an AUV. From the preliminary results, it can be deduced that a multivariable model should be used, where the couplings between depth and pitch is directly taken into account. The results confirm also the excellent numerical capability of the algorithm in the presence of intense measurement noise.

REFERENCES

Kybic, J., T. Blu and M. Unser (2001). Generalized sampling : a variational approach, *Technical Report Biomedical Imaging Group*, Swiss Federal Institute of Technology, Lausanne, Switzerland, 12 March 2001.

Juang , J.-N. , J.E. Cooper andJ.R. Wright (1988). An eigensystem realization algorithm using data correlations (ERA/DC) for modal parametersidentification, *Control Theory and Advanced Technology*,Vol4 ,pp.5-14.

Juang, J.-N. (1994). *Applied System Identification*, Prentice Hall, New Jersey.

Naeem, W., R. Sutton R., and J. Chudley (2003). System identification, modelling and control of an autonomous underwater vehicle. *Proc. MCMC2003IFAC Conference*, Girona, Spain, 17-19 September 2003.

IAV2004 - PREPRINTS
5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles
Instituto Superior Técnico, Lisboa, Portugal
July 5-7, 2004

THE APPLICATION OF SOFT COMPUTING TECHNIQUES TO AN INTEGRATED NAVIGATION SYSTEM OF AN AUV

D. Loebis*, R. Sutton*, J. Chudley*, W. Naeem*, F. R. Dalgleish, S. Tetlow****

* *Marine and Industrial Dynamic Analysis Research Group, Reynolds Building, School of Engineering, The University of Plymouth, Drake Circus, Plymouth, PL4 8AA*

** *Offshore Technology Centre, Cranfield University, Bedfordshire, Cranfield, MK43 0AL*

Abstract: This paper describes the implementation of a soft computing method based on fuzzy logic and multiobjective genetic algorithm (MOGA) techniques to adapt the parameters of an error-state complementary Kalman filter (ESCKF) to enhance the accuracy of an autonomous underwater vehicle (AUV) integrated navigation system. In the ESCKF, inertially-derived quantities from an inertial navigation system (INS) sensor are combined with direct measurements of the same quantities by use of the global positioning system (GPS) when the vehicle is on the surface and the velocity estimator output from a visual navigation system (VNS) based on laser stripe illumination methodology when the vehicle is in close contact to the bottom of the sea whilst performing an underwater mission. This strategy will alleviate the need for frequent excursions to the surface to obtain a GPS fix to reset the navigation solution produced by the INS that tends to drift after a certain period of time. This technique exploits the complementary error characteristics in such a way so as to produce optimal estimates in terms of minimum variance. To this end, errors of the sensors are modelled using first-order Markov processes and error data analysis is undertaken to determine the respective time constants and variances. To maintain the stability and performance of the ESCKF, which is likely to deteriorate when the assumed error and noise characteristics do not reflect the true ones, a fuzzy logic based scheme is used to make these values adaptive. The choice of fuzzy membership functions for this scheme is first carried out using a heuristic approach and further refined using a MOGA method.

Keywords: Autonomous underwater vehicles, navigation, visual motion, Kalman filters, fuzzy logic, genetic algorithm, multiobjective optimization

1. INTRODUCTION

A deep mobile target has been converted into a rudimentary autonomous underwater vehicle (AUV) known as *Hammerhead*. A three year co-operative research project funded by the Engineering and Physical Sciences Research Council involving both the University of Plymouth (UoP) and Cranfield University (CU) has the objectives of designing and developing an interactive navigation system consisting of a visual navigation subsystem (VNS) and an inertial navigation system/global positioning subsystem (INS/GPS) to interact with an appropriate guidance and control system. VNS has been widely adopted as a navigation methodology for AUVs as it has the capabilities to provide high precision and high quality measurements from image data to derive accurate relative position information. Advanced VNS applications can attempt to provide AUV global position updates, while simultaneously creating a mosaic, or composite image map of a seabed (Fleischer *et al.*, 1997) and matching current image with viewing mosaic map. VNS has also been used for tracking and cable following (Balasuriya and Ura, 1999). In this work various image-processing techniques to extract position measurements or to identify a specific feature of an object from live video imagery are used. Given the high resolution of digital imaging, measurement accuracies on the order of millimetres and precise feature identification can be achieved. However, the methods employed are limited in regimes where the object or terrain of interest is within both the field of view (FoV) and visual range of the camera. In addition to that, there is a need for artificial light, which increases the expense and power consumption of the vehicles. The fusion of VNS with INS measurement data can be proposed as a potential solution to these problems, as INS measurements are not affected by the aforementioned factors (Fleischer *et al.*, 1997; Balasuriya and Ura, 1999). Through the technique of dead reckoning, the position of an AUV can be inferred by integrating the fused VNS and INS measurements. The problem here is that the dead reckoning is only accurate for short time durations; since the measurement noise is integrated along with the signals, the error on position accumulates quite quickly. Consequently, an external reset mechanism is required. The use of GPS to provide periodic updates and compensate drifting from the bias errors inherent when integrating INS heading for position, have been widely implemented in the navigation of AUVs. The work in this paper is an extension to the general integrated INS/GPS by fusing the VNS and INS data between intermittent GPS fixes. The navigation system that is being developed at the UoP is based on a multisensor data fusion (MSDF) technique that can produce accurate navigation information continuously in real time from a variety of low cost inertial sensors and a GPS receiver. During an actual mission this subsystem is enhanced by data from the intelligent viewing system developed by CU, with the purpose of aiding navigation by providing velocity estimates. Once the navigation data has been suitably processed it will be fed to the guidance and control system for the appropriate action. The aim of this paper is to describe the present hardware/sensor configurations and techniques to combine measurement data from the VNS, INS and GPS to derive an estimated position of the *Hammerhead* AUV during both submerged and surface operations. The structure of this paper is as follows: the next two sections describe the current status of the *Hammerhead* VNS and INS/GPS development respectively and concluding remarks are given in section 4.

2. *HAMMERHEAD* VNS DEVELOPMENT: CURRENT STATUS

2.1 General description

The *Hammerhead* VNS is based on the laser stripe illumination (LSI) methodology previously developed at CU (Tetlow and Allwood, 1995), and will provide enhanced viewing of the seabed below the vehicle. However, it also provides real time data such as velocities, altitude and tracking information to the navigation system during the mission as well as gathering images to produce a post mission enhanced optical waterfall image of a surveyed area. There are several advantages of this type of approach over conventional imaging. LSI provides an improved image contrast at a given range. From computer simulations (Jaffe and Dunn, 1988) this type of system becomes limited at 5-7 attenuation lengths, compared with 2-4 for a conventionally illuminated system. This allows an increased deployment altitude for seabed surveys (3-18 metres) resulting in a greater swathe and hence greater area coverage. The images produced are approximately optically flat, meaning they exhibit even illumination. Furthermore, the structured nature of the light allows additional geometric information to be derived from the image and the stripe region can be extracted for each image to form a continuously evolving 2-D intensity waterfall image. However, LSI systems are more expensive and require a greater development resource than conventional systems. The viewing system comprises: a 100mW frequency doubled diode pumped Nd:YAG Laser (532nm), a low cost high sensitivity monochrome charged coupled device (CCD) camera with a wide angle lens, a single axis scanner and a tilt-compensated electronic compass (TCM2). Both the laser/scanner assembly and the camera are mounted within separate dedicated sections of the torpedo-shaped vehicle, with specially made plane ports to accommodate the optical path. The TCM2 compass uses two inclinometers to correct the output of three magnetometers for the declination angle error. The inclinometers, which are liquid-filled, are integral to the viewing system, providing tilt data for stabilisation when pitch and roll is experienced. The complete sensor subsystem can be split into the

three areas, with regard to their distinct utility: 1. velocity estimator; 2. active altitude sensor; 3. imaging capabilities.

A computer vision application is used to derive the required navigational and tracking information in real time from acquired images. An estimate of the instantaneous speed of the vehicle can be derived by using a 2-D correlation-based window-tracker, together with an integration of a range estimate from a laser-triangulation system by which image displacements are transformed into real-world displacements. The laser-triangulation technique requires the extraction of the vertical position of the laser stripe on an image to determine the range from the centre of the camera axis to the seabed. Together with the speed and range acquisition in real time, image quality can be checked and optimized by either changing the laser system parameters or demanding navigational changes from the vehicle. The complete video sequence is also recorded in digital-8 format and can be post-processed to produce a continuous 2-D intensity waterfall image or 3-D range images of the seabed with dimensional data and referenced against accurate positional information. These optical maps can be used to classify and locate particular objects that are of interest. This can be implemented in the image-processing suite or by manual inspection of the mosaic. In the autumn of 2002 a set of constrained motion trials were performed at the IFREMER facility in Brest. As well as the validation of a viewing model and system calibration, these trials created a useful archive of test files, allowing much of the future development to be possible from the dry laboratory. Furthermore, the measurements necessary to build a ground truth model were acquired in parallel to the image and vehicle specific data. This is used in ascertaining the accuracy of the vision-based navigation routines. A description of these experiments, including the construction, analysis and limitations of the ground truth model is given in a previous paper (Dalgleish *et al.*, 2003). The next sub-section briefly describe the sensor in terms of one navigational components, *i.e.* the velocity estimator. Readers interested in the second component, the active altitude sensor, are referred to Loebis *et al.* (2003*a*). Some recent results are presented and the means by which the outputs are to be integrated with the MSDF algorithm (described later in the paper) is discussed. More detail concerning each component and the imaging capabilities will be given in subsequent papers.

2.2 Velocity estimator

As an implementation of window-based tracking (Anandan, 1989) using the LabVIEW IMAQTM

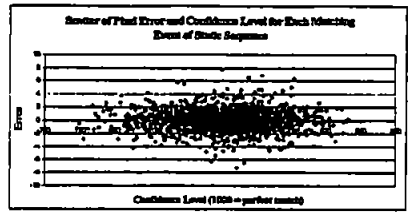


Fig. 1. Scatter plot of error vs. confidence level

Image Processing toolbox, this algorithm performs an intelligent correlation between an initial stored image region and subsequent image regions where image displacement is measured directly. The tracker uses a non-uniform sampling technique where only a few points that represent the overall content of the image are extracted. Moreover, an adaptive search strategy based on previous displacements is also incorporated to further improve tracking efficiency. Each successful matching event outputs a confidence level based on the degree of success of the correlation. Fig. 1 illustrates the distribution of matching events for the single resolution case and the corresponding pixel error, where sub-pixel accuracy is used. This was based on a static subsea sequence of two thousand 'lossy' JPEG images. The stand-off distance was eight metres. The window size was 20 × 20 pixels. As a more realistic alternative to adding Gaussian noise to each pixel independently, it is planned to use an 'artificial turbidity' environment in the Cranfield test tank to assess the static performance of the single resolution tracker under increasing noise. It is intended to use the confidence value as an indication of image quality, where as the confidence value degrades, the desired altitude of the AUV needs to be reduced. Outliers are detected in a smoothing stage.

3. *HAMMERHEAD* INS/GPS DEVELOPMENT: CURRENT STATUS

3.1 ESCKF Modelling

Brown and Hwang (1997) discuss the advantages of the ESCKF method over the total state Kalman filter. The most important advantage is that any nonlinear relationship between the process dynamics in the inertial system and the measurement relationships can be removed in a differencing operation, and the filter becomes linear. This linearity condition is required by the Kalman filter. This condition can also lead to a faster codes execution as linearisation operations are relatively slow to execute. KF algorithms are widely available in the literature. The interested reader can refer to Brown and Hwang (1997). Works on ESCKF

$$\mathbf{F} = \begin{bmatrix} 0\,0\,0 & 0 & 0 & -\sin\hat{\psi}^g\cdot\hat{u}^h - \cos\hat{\psi}^g\cdot\hat{v}^h & 0 & \cos\psi^g_m & -\sin\psi^g_m \\ 0\,0\,0 & 0 & 0 & \cos\hat{\psi}^g\cdot\hat{u}^h - \sin\hat{\psi}^g\cdot\hat{v}^h & 0 & \sin\psi^g_m & \cos\psi^g_m \\ 0\,0\,0 & 0 & 0 & 0 & \frac{cos\phi^h_m}{\cos\theta^h_m} & 0 & 0 \\ 0\,0\,0 & \frac{-1}{\tau_x} & 0 & 0 & 0 & 0 & 0 \\ 0\,0\,0 & 0 & \frac{-1}{\tau_y} & 0 & 0 & 0 & 0 \\ 0\,0\,0 & 0 & 0 & \frac{-1}{\tau_\psi} & 0 & 0 & 0 \\ 0\,0\,0 & 0 & 0 & 0 & \frac{-1}{\tau_\psi} & 0 & 0 \\ 0\,0\,0 & 0 & 0 & 0 & 0 & \frac{-1}{\tau_u} & 0 \\ 0\,0\,0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{\tau_v} \end{bmatrix} \quad (3)$$

however, are very limited especially in the field of AUV navigation systems. An example of this work can be found in Gustaffson *et al.* (2001). Like in the KF, the ESCKF algorithm can be divided into two major parts: the measurement update and the time update. In ESCKF, the measurement update is obtained by subtracting the direct measurements from the computed version of the same quantities. By doing this, the true values cancel each other out and what remains is the difference between the measurement errors and drift errors. In the time update, the estimates are obtained by subtracting the estimated drift errors from the forward filter pass from the computed version of the same quantities. In this paper, measurement errors from an accelerometer and a gyroscope (assembled in an inertial measurement unit(IMU)), a TCM2 electronic compass and a GPS receiver unit are estimated and modelled using first order Markov processes which are defined in the following manner:

$$\dot{x} = -\frac{1}{\tau}\cdot x + \gamma \quad (4)$$

In (4), x is the error process to be modelled, τ is the time constant of the assumed Markov process and γ is white noise. For modelling purposes, all sensor data have been collected in static conditions for a period of approximately 2.5 hours. For the same purpose, three different frames of reference are defined. The body-fixed (b) frame of reference is aligned to the axes of the AUV, where forward-starboard-down correspond to x - y - z. These need to be transformed to the geographical (g) frame of reference, where x - y - z correspond to North-East-Down. For these particular application, the measurements in question are 3D accelerations, as well as angular rates measured by the IMU. Earth-centred Earth-fixed (ECEF) frame is where the GPS latitude and longitude are defined. The elements of the state of the ESCKF are defined as follows:

$$\mathbf{x} = \left[\, x^g_d \; y^g_d \; \psi^g_d \; x^g_e \; y^g_e \; \psi^g_e \; r^b_e \; u^h_e \; v^h_e \,\right] \quad (5)$$

In (5) the subscripts d and e denote drift and sensor errors respectively. Superscripts g, b and h denote geographical frame, body-fixed-frame and horizontal frame respectively. Drift errors in position, x^g_d and y^g_d stem from the error in the integrated acceleration (u^h_e and v^h_e), and compass error ψ^g_e. ψ^g_d is heading drift error which comes from the error in the integrated yaw rate. Measurement errors in the position bland are respectively represented by the states x^g_e and y^g_e. Finally, state r^b_e represents gyroscope's yaw error. The process matrix (F) is given in (3). In (3), m denotes measured value. Due to page limitation, the derivation of this matrix is not given in this paper.

The variance of the process noise for a Markov error model can be described as in Brown and Hwang (1997),

$$variance[w_k] = (1 - e^{-\frac{2\Delta t}{\tau}})\cdot variance[x_k] \quad (6)$$

In (6), $e^{-\frac{\Delta t}{\tau}}$ is the state transition parameter for the Markov error model. Δt is the discrete time interval and τ is the time constant. By taking the approximation: $e^{-\beta\Delta(t)} \approx 1 - \beta\cdot\Delta t$, where $\beta = \tau^{-1}$, the following is true:

$$variance[w_k] = (2\beta\Delta t - (\beta\Delta t)^2)\cdot variance[x_k] \quad (7)$$

According to the process model, the heading drift error state represents the integrated yaw error state, in effect an integrated Markov process. The process noise covariance matrix for these two states can be defined as in Brown and Hwang (1997),

$$\mathbf{Q} = \begin{bmatrix} E[\psi^g_d\psi^g_d] & E[\psi^g_d r^b_d] \\ E[\psi^g_d r^b_d] & E[r^b_d r^b_d] \end{bmatrix} \quad (8)$$

where

$$Q_{11} = \frac{2\sigma^2}{\beta}\left[\Delta t - \frac{2}{\beta}(1-\phi) + \frac{1}{2\beta}(1-\phi^2)\right] \quad (9)$$

$$Q_{12} = Q_{21} = 2\sigma^2\left[\frac{1}{\beta}(1-\phi) + \frac{1}{2\beta}(1-\phi^2)\right] \quad (10)$$

$$Q_{22} = \sigma^2(1-\phi^2) \quad (11)$$

where β is the inverse of the Markov time constant, σ^2 is the process noise variance of the yaw rate Markov error and $\phi = e^{-\frac{\Delta t}{\tau}}$ and defined as before.

For the first two states in the process model, x_d^g and y_d^g, the analysis is more complicated and for this reason, the noise covariance matrix of the two states are obtained from an empirical result, and provision for the adjustment method has been made and will be reported in the future.

The measurement matrix $\mathbf{H}$ relates the available measurement updates to the element in the state vector and takes the following form:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad (12)$$

The measurement noise covariance matrix $\mathbf{R}_k$ is determined empirically and given as:

$$\mathbf{R}_k = \begin{bmatrix} \sigma^2_{X-Position} & 0 & 0 \\ 0 & \sigma^2_{Y-Position} & 0 \\ 0 & 0 & \sigma^2_{Heading} \end{bmatrix} \quad (13)$$

where $\sigma^2_{X-Position}$, $\sigma^2_{Y-Position}$, $\sigma^2_{Heading}$ are the variance in X,Y direction and heading respectively. These values will be adapted using the algorithm discussed in the next sub-section.

3.2 Fuzzy error state complementary Kalman filter (FESCKF)

In this sub-section, an on-line innovation-based adaptive scheme of the ESCKF to adjust the $\mathbf{R}$ matrix employing the principles of fuzzy logic is presented. The fuzzy logic is chosen mainly because of its simplicity. This motivates the interest in the topic, as testified by related papers which have been appearing in the literature (Loebis *et al.*, 2003*b*; Escamilla-Ambrosio and Mort, 2001). The FESCKF proposed herein is based on the IAE approach using a technique known as covariance-matching (Mehra, 1970). The basic idea behind the technique is to make the actual value of the covariance of the innovation sequences match its theoretical value.

The actual covariance is defined as an approximation of the $\mathbf{Inn}_k$ sample covariance through averaging inside a moving estimation window of size M which takes the following form:

$$\hat{\mathbf{C}}_{Inn_k} = \frac{1}{M}\sum_{j=j_0}^{k} \mathbf{Inn}_k \cdot \mathbf{Inn}_k^T \quad (14)$$

where $j_0 = k - M + 1$ is the first sample inside the estimation window. An empirical experiment is conducted to choose the window size M. From experimentation it was found that a good size for the moving window in (14) is 15. The theoretical covariance of the innovation sequence is defined as

$$\mathbf{S}_k = \mathbf{H}_k \cdot \mathbf{P}_k^- \cdot \mathbf{H}_k^T + \mathbf{R}_k \quad (15)$$

The logic of the adaptation algorithm using covariance matching technique can be qualitatively described as follows. If the actual covariance value $\hat{\mathbf{C}}_{Inn_k}$ is is observed, whose value is within the range predicted by theory $\mathbf{S}_k$ and the difference is very near to zero, this indicates that both covariances match almost perfectly and only a small change is needed to be made on the value of $\mathbf{R}$. If the actual covariance is greater than its theoretical value, the value of $\mathbf{R}$ should be decreased. On the contrary, if $\hat{\mathbf{C}}_{Inn_k}$ is less than $\mathbf{S}_k$, the value of $\mathbf{R}$ should be increased. This adjustment mechanism lends itself very well to being dealt with using a fuzzy-logic approach based on rules of the kind:

$$IF\ (antecedent)\ THEN\ (consequent) \quad (16)$$

where antecedent and consequent are of the form $\nu \epsilon O_i$, $\kappa \epsilon L_i$, $i = 1, 2, ...$ respectively. Where ν and κ are the input and output variables, respectively, and O_i and L_i are the fuzzy sets.

To implement the above covariance matching technique using the fuzzy logic approach, a new variable called $\mathbf{delta}_k$, is defined to detect the discrepancy between $\hat{\mathbf{C}}_{Inn_k}$ and $\mathbf{S}_k$. The following fuzzy rules of the kind (16) are used:

$$IF\ (\mathbf{delta}_k \cong 0)\ THEN\ (\mathbf{R}_k\ is\ unchanged) \quad (17)$$

$$IF\ (\mathbf{delta}_k > 0)\ THEN\ (\mathbf{R}_k\ is\ decreased) \quad (18)$$

$$IF\ (\mathbf{delta}_k < 0)\ THEN\ (\mathbf{R}_k\ is\ increased) \quad (19)$$

Thus $\mathbf{R}$ is adjusted according to,

$$\mathbf{R}_k = \mathbf{R}_{k-1} + \Delta\mathbf{R}_k \quad (20)$$

where $\Delta\mathbf{R}_k$ is added or subtracted from $\mathbf{R}$ at each instant of time. Here $\mathbf{delta}_k$ is the input to the fuzzy inference system (FIS) and $\Delta\mathbf{R}_k$ is the output.

3.3 Fuzzy membership functions optimisation

MOGA is used here to optimize the membership functions of the FKF. To translate the FKF membership functions to a representation useful as genetic material, they are parameterised with real-valued variables. Each of these variables constitutes a gene of the chromosomes for the MOGA. Boundaries of chromosomes are required for the creation of chromosomes in the right limits so that the MOGA is not misled to some other area of search space. The technique adopted in this paper is to define the boundaries of the output membership functions according to the furthest points and the crossover points of two adjacent membership functions. In other words, the boundaries of FKF consist of three real-valued chromosomes (*Ch*s), as in Figure 2. The trapezoidal membership

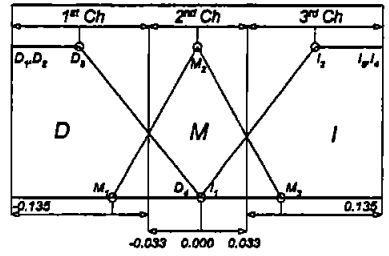


Fig. 2. Membership function and boundaries of $\mathbf{R}_k$

functions' two furthest points, -0.135 (D_1), -0.135 (D_2) and 0.135 (I_3), 0.135 (I_4) of FKF, remain the same in the GA's description to allow a similar representation as the fuzzy system's definition. As can be seen from Figure 2, D_3 and M_1 can change value in the 1st *Ch* boundary, D_4, M_2 and I_1 in the 2nd *Ch* boundary, and finally, M_3 and I_2 in 3rd *Ch*.

4. CONCLUDING REMARKS

A novel method to obtain an accurate AUV navigation system is proposed. The method is based on the ESCKF coupled with fuzzy logic to adjust the value of measurement noise covariance matrix $\mathbf{R}$ during both surface and underwater mission whereby different combination of sensors, i.e. INS/GPS and INS/VNS respectively, are used. MOGA is proposed to further refine the result.

REFERENCES

Anandan, P. (1989). A Computational Framework and an Algorithm for the Measurement of Visual Motion. *International Journal of Computer Vision* **2**, 283–310.

Balasuriya, B. A. A. P. and T. Ura (1999). Sensor Fusion Technique for Cable Following by Autonomous Underwater Vehicles. In: *Proceedings of the 1999 IEEE International Conference on Control Applications*. Kohala Coast-Island of Hawaii, Hawaii, USA. pp. 1779–1784.

Brown, R. G. and P. Y. C. Hwang (1997). *Introduction to Random Signals and Applied Kalman Filtering*. 3rd Ed. John Wiley and Sons.

Dalgleish, F. R., S. Tetlow and R. L. Allwood (2003). Preliminary Experiments in the Development of a Laser-based Imaging Sensor for AUV Navigation. In: *Proceedings of the 1st IFAC Workshop on Guidance and Control of Underwater Vehicles*. Newport, South Wales, UK. pp. 239–244.

Escamilla-Ambrosio, P. J. and N. Mort (2001). A Hybrid Kalman Filter-Fuzzy Logic Multisensor Data Fusion Architecture with Fault Tolerant Characteristics. In: *Proc. of the 2001 International Conference on Artificial Intelligence*. Las Vegas, NV, USA. pp. 361–367.

Fleischer, S. D., M. R. Rock and R. Burton (1997). Global Position Determination and Vehicle Path Estimation from a Vision Sensor for Real Time Video Mosaicking and Navigation. In: *Proceedings of the Oceans '97 MTS/IEEE*. Halifax, Nova Scotia, Canada. pp. 641–647.

Gustaffson, E., E. An and S. Smith (2001). A Postprocessing Kalman Smoother for Underwater Vehicle Navigation. In: *Proc. 12th International Symposium on Unmanned Unthetered Submersible Technology*. New Hampshire, NH, USA. pp. 1–7.

Jaffe, J. and C. Dunn (1988). A Model-Based Comparison of Underwater Imaging Systems. *SPIE Ocean Optics IX* **925**, 344–350.

Loebis, D., F. R. Dalgleish, R. Sutton, S. Tetlow, J. Chudley and R. L. Alwood (2003*a*). An Integrated Approach in the Design of a Navigation System for an AUV. In: *Proc. 6th IFAC Conference on Manoeuvring and Control of Marine Craft*. Girona, Spain. pp. 329–334.

Loebis, D., R. Sutton and J. Chudley (2003*b*). A Fuzzy Kalman Filter for Accurate Navigation of an Autonomous Underwater Vehicle. In: *Proc. 1st IFAC Workshop on Guidance and Control of Underwater Vehicles*. Newport, South Wales, UK. pp. 161–166.

Mehra, R. K. (1970). On the Identification of Variances and Adaptive Kalman Filtering. *IEEE Transactions on Automatic Control* **AC-16(1)**, 12–21.

Tetlow, S. and R. L. Allwood (1995). Development and Applications of a Novel Underwater Laser Illumination System. *Underwater Technology* **21**, 13–20.